AFRL-RI-RS-TR-2007-281
Final Technical Report
January 2008

PRIME: A PMESII (POLITICAL, MILITARY, ECONOMIC, SOCIAL, INFRASTRUCTURAL AND INFORMATIONAL) MODEL DEVELOPMENT ENVIRONMENT

SRI International

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

NOTICE AND SIGNATURE PAGE

AFRL-RI-RS-TR-2007-281 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE DIRECTOR:

/s/	/s/
DALE W. RICHARDS Work Unit Manager	JAMES W. CUSACK Chief, Information Systems Division Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
JAN 2008

2. REPORT TYPE
Final

3. DATES COVERED *(From - To)*
Mar 06 – Jun 07

4. TITLE AND SUBTITLE

PRIME: A PMESII (POLITICAL, MILITARY, ECONOMIC, SOCIAL, INFRASTRUCTURAL AND INFORMATIONAL) MODEL DEVELOPMENT ENVIRONMENT

5a. CONTRACT NUMBER
FA8750-06-C-0071

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
62702F

6. AUTHOR(S)

Ian Harrison

5d. PROJECT NUMBER
558S

5e. TASK NUMBER
CP

5f. WORK UNIT NUMBER
E9

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
SRI International
333 Ravenswood Ave
Menlo Park CA 94025-3453

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFRL/RISE
525 Brooks Rd
Rome NY 13441-4505

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-RI-RS-TR-2007-281

12. DISTRIBUTION AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED. PA# WPAFB-07-0711

13. SUPPLEMENTARY NOTES

14. ABSTRACT
As a key component of effects-based operations (EBO) is to allow analysts to examine the effects that different courses of action would have on a situation from multiple viewpoints, PRIME is a software tool for political, military, economic, social, infrastructural, and informational (PMESII) modeling that supports this need by looking beyond the traditional narrow horizon of examining only the military effects of planned actions. PRIME supports modeling of military actions, as well as diplomatic, information, and economic actions. This effort looked at the expected use cases for a PMESII modeling tool to help identify requirements. An existing structured-argument modeling tool, SEAS, was utilized as the underlying technology base, and the design modified to meet the identified software requirements. This report describes the software design and functionality of the resulting tool. Future research activities that would test the usefulness of PRIME are also discussed, with emphasis on experiments to predict higher-order effects of actions, rather than just direct effects.

15. SUBJECT TERMS
PMESII, Framework, Interoperability, Modeling, Model Integration, Commander’s Predictive Environment (CPE)

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					Dale W. Richards
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	UL	49	**19b. TELEPHONE NUMBER (*Include area code)***
U	U	U			N/A

Table of Contents

List of Figures

Executive Summary

The PRIME project set out to develop a key component of the Air Force Research Laboratory (AFRL) Commander's Predictive Environment (CPE) program, focusing on the area of *Understanding the Battlespace.* The program had identified system-of-system modeling across the political, military, economic, social, infrastructural, and informational (PMESII) dimensions as a central theme. There existed a lack of technological aid to support PMESII model development, adaptation, and maintenance. In addition, a goal of the CPE program was to find ways to rapidly create these models. SRI's PRIME software is tailored specifically to this PMESII modeling task, as it is exploited as part of an overall effects-based operations (EBO) approach.

SRI had previously developed SEAS,[1] a graphical multidimensional strategic decision-making tool for use by the Intelligence Community (IC), which is in operational use at a number of IC sites. SEAS is a general-purpose structured argumentation tool that allows for the collaborative development of multidimensional models by a team of analysts who may be separated in time and space. Our vision for PRIME was to leverage this existing Department of Defense (DoD) investment and develop a tailored modeling interface for PMESII modeling. Our aim was to reuse many of the visualizations utilized by SEAS, but to enhance these for the specifics of the PMESII modeling task. Indeed, SEAS had been previously used as part of a Joint Forces Command (JFCOM) Multinational Experiment (MNE'02) to conduct experiments in collaboratively developing PMESII models (POC: Donald C. Greaser, J9DIA C2). This experiment had demonstrated that SEAS could produce the required PMESII models, but that the SEAS graphical user interface (GUI) did not adequately support the end-to-end PMESII modeling life cycle.

The work initially involved examining the actual concept of operations (CONOPS) for PMESII modeling in order to develop requirements for the PRIME tool. The work looked at the document produced by the United States Joint Forces Command, *Doctrinal Implications of Operational Net Assessment (ONA)* (http://www.dtic.mil/doctrine/education/jwfc_pam4.pdf). Taking these requirements into account, it was then determined what new or enhanced functionality was required, above and beyond that provided by SEAS, in order that PRIME would fully support end-to-end PMESII modeling. With the requirements and design work completed, technical implementation of PRIME was begun. PRIME was developed as a plug-in to the Government distribution of SEAS (Version 7.1). This meant that the full underlying capability of SEAS is actually available within PRME, but is not exposed in the actual PRIME user interface. This architectural decision meant that modifications or extensions to PRIME could draw on this base SEAS capability, if necessary or desired at a later date. At the same time, much of the complexity of SEAS and generality was viewed as being nonessential to PMESII analysts and so was hidden from view.

[1] SEAS – http://www.seas.sri.com

Two versions of PRIME were developed. Version 1.0 at 6 months into the project was a proof-of-concept demonstrator designed to explore how best to exploit SEAS's underlying capabilities and graft a tailored GUI on top, while at the same time introducing many of the concepts and processes for PRIME's envisioned PMESII modeling CONOPS. Version 2.0, released at the completion of the project and delivered to AFRL, is a comprehensive PMESII modeling tool, which fulfills the original goals as set out in the PRIME project proposal. PRIME is a web-server-based application, with the client being a modern browser (we tested primarily on Firefox/Mozilla), eliminating the need for client installation on each user desktop. SRI also stood up a live server at its Menlo Park headquarters to allow AFRL and external evaluators to test and evaluate PRIME without having to install the PRIME server at their own sites. PRIME comes with an online help system.

We did not fully explore all that we set out to do as part of the program because of resource limitations and redirection by CPE program management. In particular, we had intended to explore effects-based predictions for *indirect* (second- and higher-order effects) as well as direct effects. Our aim was to leverage SRI's Link Analysis Workbench (LAW[2]) developed under the ARDA EAGLE program. This experimental exploration work was never conducted. Despite this, the overall effect was that our project became more focused on our central goal, which was the development of a purely direct effects PMESII modeling tool.

The PRIME project successfully delivered a fully functioning PMESII modeling tool was delivered, which had capabilities beyond our original proposal. PRIME's worth will become apparent only when it is evaluated by potential end users to determine whether it fulfills their technical needs and provides cost-effective modeling support beyond what current technology provides. In many cases, current technology is still paper and pen based, in a BOGSAT (bunch of guys sitting around a table) mode. PRIME allows people to collaboratively develop PMESII models, even when they are based in different locations. These models are shareable and editable, and allow for the recording of team discussions about model accuracy. We expect that PMESII model development using PRIME will be found to be much more rapid than traditional methods. Model quality; adaptability, and ease of maintenance will provide an added bonus.

[2] LAW – http://www.ai.sri.com/~law

1. Introduction

This final technical report documents work undertaken by SRI as part of the Air Force Research Laboratory (AFRL) Commander's Predictive Environment (CPE) program. SRI's proposal to this program was to address a key problem identified by the program: the lack of rapid PMESII modeling development tools to support PMESII modeling in an effects-based operations (EBO) setting. SRI proposed developing a software prototype, named PRIME, which would fully support direct-effects PMESII modeling.

The envisaged PRIME software was not to be developed from scratch. Instead, it was to be a tailored plug-in to an existing investment, SEAS,[3] developed by SRI over a number of years. SEAS is a general-purpose structured argument modeling tool. SEAS had been previously used as part of a Joint Forces Command (JFCOM) Multinational Experiment (MNE'02), to conduct experiments in collaboratively developing PMESII models (POC: Donald C. Greaser, J9DIA C2). This experiment demonstrated that SEAS could produce the required PMESII models, but that in its existing form it was not ideally suited to this task because of its general-purpose modeling nature.

The project conducted a brief use case analysis, which led to clearer and tighter design requirements than had been envisaged in the original proposal. Taking these design requirements into account, the PRIME software was developed and delivered to AFRL. The PRIME distribution included installation and configuration documentation, as well as an extensive user manual. In addition, SRI stood up a live PRIME server to allow users to trial PRIME without having to install the PRIME software at their sites. PRIME is a client-server web application, with its server running on a Sun Solaris server. The client must be running either Firefox or Mozilla browsers.

This report details the use case and design work conducted on the project and describes the developed functionality of PRIME in detail, focusing on its use for the PMESII modeling task. Future research directions and conclusions also are reported.

2. PRIME CONOPS

The overall goal of using PRIME is to produce a global predicted direct effects PMESII model for a planned set of actions on a site. Actions for PRIME PMESII modeling are diplomatic, informational, military, or economic (DIME). A site is either an entity (*node* in PMESII modeling terminology) or set of entities. An entity can be a person/organization, a physical entity, e.g., geographical such as region/country/village or infrastructural such as a building/road, a conceptual entity, e.g., a religious icon or a cultural icon, or a sector of the economy, e.g., the banking industry.

[3] SEAS –see http://www.seas.sri.com

Three primary use cases are envisaged for PRIME.

1) In an installation phase of PRIME at an organization, the PMESII model template, on which all PMESII modeling is done, can be tailored to the needs of the organization. PRIME is delivered with an example PMESII model template, but this template can be edited as necessary. In addition, PRIME is delivered with example taxonomies of DIME actions and node types, based in part on taxonomies supplied by ISX to the CPE program. These taxonomies are editable within PRIME, allowing an organization to tailor the action and node types to its specific needs.

2) In a pre-crisis setting, PRIME is used to develop a library of *generic* PMESII models. These generic models record an analyst's predictions of the effect that a single **type** of action would have on a single **type** of node. An example might be the PMESII model predicting the effects that an economic embargo would have on a country. Here the actual actions that constitute what is actually done to create an economic embargo are not stated, nor is the actual country stated. Hence, the model is termed *generic.* These generic models draw on the model template and taxonomies developed in the first use case. Two other important tasks can be undertaken in pre-crisis mode (and can be continued during the in-crisis mode): site model and plan creation. A **site model** contains the entities that are being considered to have actions taken on them. A **plan** is a set of actions on entities. Plans constitute different alternative courses of action that are being considered to achieve a desired goal.

3) In a crisis setting, PRIME is used to develop actual predicted effect PMESII models for a planned set of actions. Given a defined plan, PRIME can then be tasked to automatically generate an overall predicted effect PMESII model for the plan. These PMESII models are in fact collections of models, one model for each entity (node) that has actions taken on it in the plan. In this way the effect of the collective sets of plan actions are separately viewable. These generated models are then editable, allowing analysts to collaborate and reach final consensus on the overall predicted effect PMESII model.

Given these use cases the requirement for PRIME was to provide graphical user interface (GUI) support for all three use cases.

3. PRIME Software Design

PRIME was designed as a plug-in to SRI's SEAS[4] structured argumentation tool. By *plug-in* we mean that PRIME is developed on top of a standard SEAS distribution, but with extra specific code for PRIME loaded in to provide the desired PRIME functionality. SEAS is a powerful and flexible structured argumentation tool that had been used previously in a Joint Forces Command (JFCOM) Multinational Experiment (MNE'02) to conduct experiments in collaboratively

[4] SEAS – http://www.seas.sri.com

developing PMESII models. The end product was a useful PMESII model, but the process used to get to this end product was complex and required having a strong grasp of SEAS functionality.

For PRIME it was felt that a more tailored GUI should be developed to support the specific needs of PMESII modeling, hiding SEAS functionality details that were irrelevant to PMESII modeling and providing complete support for the whole PMESII modeling life cycle, as defined in the three use cases outlined in Section 2. In addition, underlying extensions to the SEAS software were needed to account for PRIME's specific requirements.

3.1 PRIME Back End Design

The major change required for the back end was to extend SEAS's underlying object class hierarchy to include concepts that are undefined in SEAS. SEAS has the concept of a model template, a multidimensional model, and a collection, all of which are utilized in PRIME. A PRIME PMESII model template is simply a specific instance of a SEAS multidimensional template, and inherits all features from SEAS. A PMESII model is a SEAS multidimensional model, which uses a PMESII-specific template, i.e., this template has six dimensions relating to the political, military, economic, social, infrastructural, and informational dimensions in all PMESII models. A specific PMESII model – one that predicts the effects of a plan on a site – is realized as a SEAS collection of individual PMESII models, one for each node referenced in the plan.

SEAS, however, lacked any concept of a plan, plan element, action, site model, or node. Hence, the underlying class hierarchy of SEAS had to be extended to incorporate these fundamental concepts. A plan in PRIME is defined as a subclass of the SEAS collection class, where the collection items are specialized to be of type plan element. A PRIME plan element is a triple of an action against a node that is part of a particular site model. A plan action in PRIME is defined as being of a certain type, where the type is from the DIME action taxonomy used by PRIME. A site model is defined as a subclass of the SEAS collection object – a site model consists of a collection of items, specialized to being nodes. A site node in PRIME is defined as being of a certain type, where the type is from the node type action taxonomy used by PRIME.

In addition to extending SEAS's original underlying class hierarchy, it became apparent that allowing organizations to use their own action type and node type taxonomies within PRIME was crucial. Hence, a taxonomy editor (see Section 3.2) was developed as part of PRIME. In addition, code was developed to allow OWL encoded taxonomies (http://www.w3.org/TR/owl-features/) to be loaded into PRIME. This capability allows organizations to import node type and action type taxonomies they may have previously defined into PRIME, without having to recreate these from scratch. This OWL file load capability is utilized by PRIME itself, in that the PRIME distribution comes with a default set of node and action types in an OWL taxonomy, which is loaded when the PRIME server software starts.

3.2 PRIME GUI Design

Driven by the requirements outlined in Section 2 the following new GUI design elements were deemed necessary for PRIME, above and beyond that provided by SEAS:

1) Improved Manager Window. In SEAS, the main window where the user centers work was a generic explorer-like interface. For PRIME it was decided to design a more PMESII modeling-process-oriented interface.
2) Taxonomy Editor. SEAS did not contain a built-in taxonomy editor. Instead, SEAS is delivered with a fixed set of object meta-data. For PRIME it was felt that allowing organizations to tailor the taxonomies used within the PMESII modeling process would be very advantageous.
3) Site Editor. Site models are specific to PMESII modeling and as such there was no existent equivalent class of object in SEAS. Hence, a site editor was critical for PMESII modeling.
4) Plan Editor. Like site models, plans are specific to PMESII modeling and as such there was no existent equivalent class of object in SEAS.
5) Generic Model Editor. SEAS provides several different structured argument model editor and viewer interfaces, but the most useful and compact for PMESII modeling purposes was available only as a viewer. It was decided to modify this viewer to add simple editing capabilities, specific to PMESII modeling, and to remove from the viewer interface SEAS elements that were irrelevant for PMESII modeling
6) PMESII Model Editor. PMESII models are in general collections of models, each model showing the predicted effects on a single entity (*node* in PMESII terminology). Hence, these are termed *node effect models*. SEAS provides a *collections editor*, which allows sets of models to be grouped together but it is missing certain information necessary for a PMESII model editor. The base SEAS collection editor was modified slightly to provide what was necessary, and to remove from the editor interface SEAS elements that were irrelevant for PMESII modeling
7) PMESII Node Effect Model Editor. The node effect model editor is essentially a superset of the generic model editor, with equivalent functionality. That is, it allows analysts to collaborate and refine the auto-generated predicted PMESII model, based on planned actions on that node. It differs from the generic model editor in the meta-data that is exposed. This meta-data identifies the overall PMESII model that the node effect model is part of, the specific actions in the plan that have been carried out, and the set of generic models that have been merged to produce the node effect model.

4. PRIME Software Implementation

4.1 SEAS Software

As mentioned previously, PRIME was developed as a plug-in to SRI’s SEAS software. Two different versions of SEAS are available for distribution. The Government off-the-shelf (GOTS) version of SEAS, known as *GOTS SEAS*, is available for U.S. Government use free of any license fees. *High SEAS*, the commercial off-the-shelf (COTS) version, is available under license, and has associated fees. PRIME was developed on top of Version 7.1 of GOTS SEAS, meaning that it could be distributed free of any license fees for U.S. Government use.

SEAS is implemented as a web server that supports the construction and exploitation of a corporate memory filled with analytic products, methods, and their interrelationships, indexed by the situations to which they apply. Objects from this corporate memory are viewed and edited through the use of a standard browser client, with the SEAS server producing ephemeral HTML based upon the contents of the SEAS knowledge base that constitutes corporate memory. The foundation of this corporate memory is an ontology of arguments and situations that includes three main types of formal objects: argument templates, arguments, and situation descriptors. Roughly speaking, an argument template records an analytic method as a hierarchically structured set of interrelated questions, an argument instantiates an argument template by answering the questions posed relative to a specific situation in the world, and situation descriptors characterize the type of situations for which the argument templates were designed and the specific situations that arguments address.

The high-level architecture of SEAS is pictured in Figure 1. SEAS consists of the SEAS Server connected to the SEAS Corporate Memory. The SEAS Server is a web server that takes in HTTP and produces HTML based upon the contents of the Corporate Memory.

Figure 2 shows more detail about the SEAS architecture. The SEAS server is built on top of the AllegroServe web server developed by Franz Inc. (http://www.franz.com). This is an HTTP server written in Common Lisp. Because the other software components underlying SEAS are SRI owned and are also written in Common Lisp, this provided an ideal basis for the development of the SEAS server. The HTML Generator creates HTML pages on the fly upon request, utilizing SRI’s ALP (Active Lisp Pages) scripting environment for creating dynamic pages and SRI’s Grasper for creating graphical depictions of SEAS arguments and templates. AllegroServe parses the HTTP requests and calls upon the HTML Generator to create pages that are then returned to the requesters by AllegroServe. The pages are generated by consulting the SEAS Corporate Memory Manager, which in turn consults the SEAS Corporate Memory stored in the Ocelot Knowledge Based Management System (KBMS). If the request changes the answer to some question in an argument, the Gister Engine is invoked to update related answers in that argument in the knowledge base. The Ocelot/Perk KBMS server has OKBC as its application programming interface (API). It optionally connects to a database management system (DBMS) server via SQL; otherwise, Ocelot stores the KB as a file in the system file.

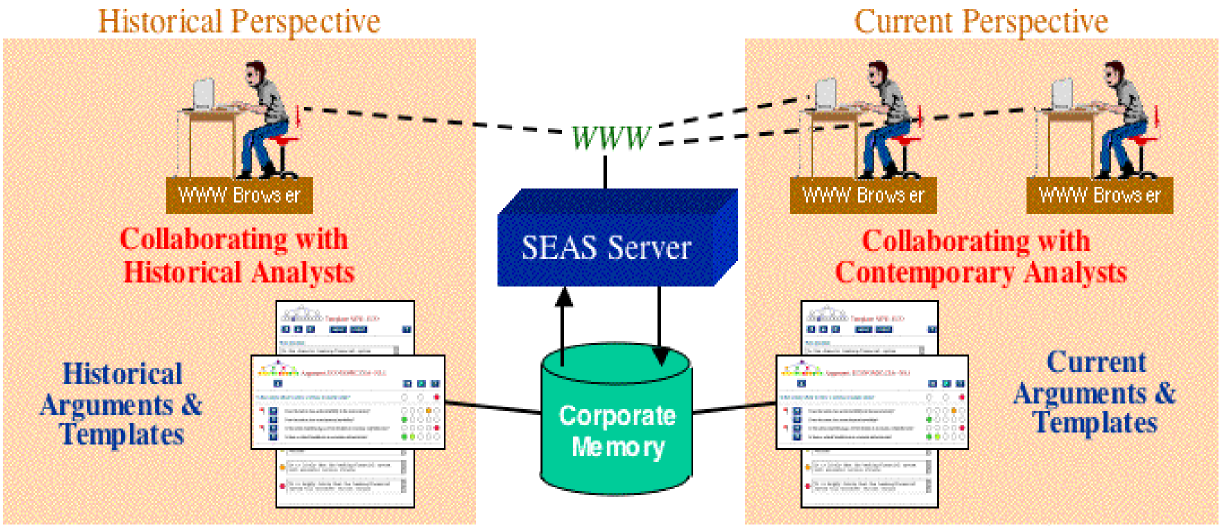

Historical Perspective
Current Perspective
WWW
WWW Browser
WWW Browser
WWW Browser
Collaborating with Historical Analysts
SEAS Server
Collaborating with Contemporary Analysts
Historical Arguments & Templates
Corporate Memory
Current Arguments & Templates

Figure 1: High-Level SEAS Architecture

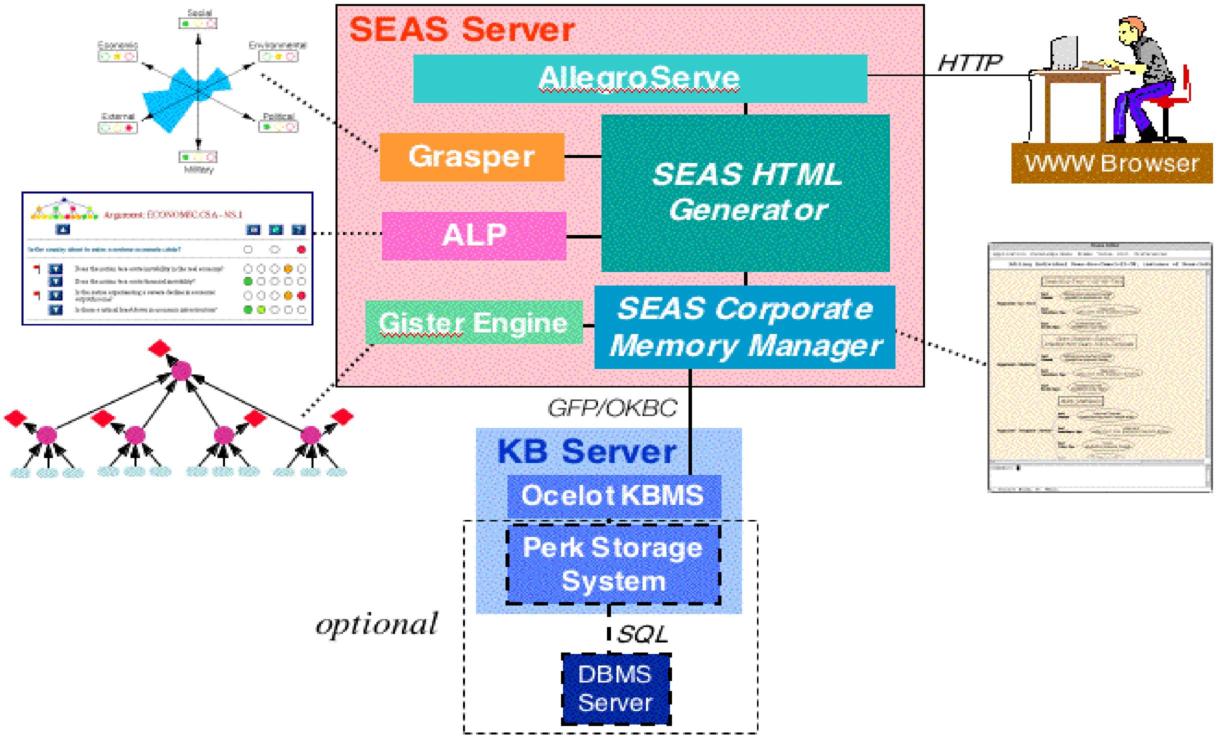


Figure 2: Detailed SEAS Architecture

4.2 PRIME Fundamental Concepts

PRIME inherits, from SEAS, some fundamental concepts that are not intuitive to all users, particularly those from a Windows application background. SEAS operates in many ways more like a database system, or like many modern web-based applications, which have a database back end.

4.2.1 User Accounts

Every PRIME user has a user account and this account can belong to one or more named PRIME user **groups**. The PRIME system administrator is the only person who can add/modify/deactivate user accounts and can add/modify/delete groups and their membership. Users of PRIME must log in to the server, like logging in to an online account, with a username and password. Hence, the initial PRIME web page that all users see is the login window. A user, who has been authenticated via an internal password check, is taken to the PRIME main window. Note that PRIME, like SEAS, does not integrate with existing LDAP servers for organizational user account information. A PRIME server must be populated with its own set of user accounts and group hierarchies.

4.2.2 Access Control

All data in PRIME has associated access control. This means, for instance, that model creators can specify who can edit as well as view a model. Access control is built around two basic concepts.

1) **Author**. A user who has read-write access is considered to be an author of an object. An author for an object can be just a single user, more than one user, a single named group of users, or a mixture of groups and users.
2) **Audience**. A user who has been given only read access to an object is considered to be a member of the object's audience.

The total set of PRIME objects available to a user is determined by the user account information (the user's group membership) together with the access control settings for each model or template. In PRIME, every plan, site model, model, or template viewer has a button, ▤ for the PRIME access control meta-data browser/editor.

4.2.3 Collaboration Support

PRIME is a web-based application that allows analysts, who could be physically separate in space and time, to work on the same PMESII effect model template, generic model, or specific model. So that analysts can work together, PRIME provides some simple facilities for two main class of collaboration support: memos and visual cues.

PRIME allows memos (akin to Post-it® note) to be left anywhere in a template or model, so that analysts can leave notes for collaborators. This can be to give feedback, to provide action items, to allow discussion, and so on. Memos are modeled on email and allow the set of people for whom a memo is applicable to be stated. As in email, users can reply to, create, and delete memos.

PRIME has two basic collaborative cues:

1) Cue to let the user know that a new (to the user) memo has been added to a template or model

2) Cue to let the user know that a new piece of evidence has been added to a model – in the form of a flag that the user can dismiss after seeing the evidence

4.2.4 Taxonomy Management

Throughout the model creation process are several places where contextual domain-relevant data (meta-data) is used. Examples include the Node and Actions types used to classify a particular PMESII model. Rather than just using free-form natural language text for this meta-data, PRIME uses taxonomies of domain terms. The advantage of using taxonomy is that there is a defined meaning for the term and that different people can apply the same term and mean the same thing by it. This in turn facilitates the retrieval or finding of models through use of these terms, for example, using the **Search** capability in PRIME.

PRIME contains a relatively simple, specialized taxonomy editor that allows organizations to define the set of PMESII-specific meta-data that they want to use for modeling. PRIME adopts the ISX CPE core ontology for its baseline terms, such as Node and Action, and all subsequent terms are sub-classed off these root terms. Hence, any imported OWL taxonomy needs to at least make reference to these core ontologies, too, in order for the PRIME ontology editor to make use of them.

4.2.5 Models vs. Templates

All PRIME PMESII effects models are created using an underlying *template*. A template provides the basic structure to the model, classifies the potential set of effects for the model, and describes the polarity of effects that an action might have on a node. PRIME can be used to create PMESII model templates specific to a particular organization or contextual need.

A PMESII effects model captures what a particular user, set of users, or group of users believes are the actual direct effects caused by applying actions to a node. The set of effects and the polarity choices come from the associated (underlying) model template. The actual selected effects (plus effect polarities) reflect what the users believe are the effects for the current context.

4.2.6 Saving Models

PRIME has no concept of a "Save", as a web-based server application all edits are made immediately in the server memory. However, the PRIME server does itself auto-save the server data out to file. Autosaving occurs only if the server data has changed in any way. The autosave files are date and time stamped. The autosave interval is by default set to every 15 minutes, but this autosave interval is configurable, as part of the PRIME server configuration, as described in the PRIME Software Release Instructions documentation that is included in the PRIME software distribution.

If the PRIME server were to crash, and the server data somehow got corrupted, then backups would be available of the server data (the auto-saved data). PRIME inherits, from SEAS, the functionality to prune these auto-save files. The auto-save cleanup policy is to retain all auto-saves for the previous 7 days, a single daily backup (the last auto-save for the day) for every day in the last 3 months (apart from the last 7 days), and a single monthly backup (the last auto-save for the month) for auto-save files older than 3 months.

4.3 PRIME Software Basics

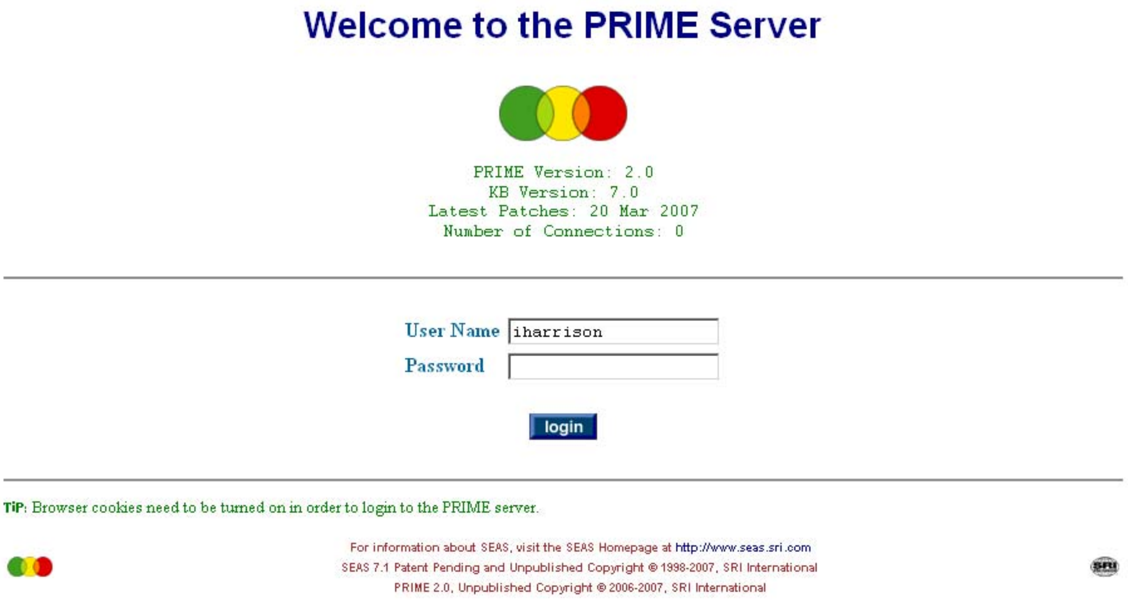
The basics of logging onto a PRIME server and orientating the reader about the main PRIME window are described here. The subsequent section is structured to reflect the three styles/phases of use identified in Section 2: installation/tailoring to an organization, pre-crisis use, and in-crisis use.

4.3.1 Logging into a PRIME server

With the PRIME software installed, configured and started, a user who has been granted an account on the server can begin using PRIME by logging in, as follows:

1) At the URL location text area of the browser you can launch PRIME by typing http://machinename.domainname:port, e.g., http://salinas.ai.sri.com:8080, using the port number that the server is working off (this is reported when the PRIME server is started). If your machine does not have a domain name but does have an IP address, use the IP address instead of the machine domain name part of the call above. If your machine does not have a domain name, i.e., DNS turned off, and you do not know its IP address, just use the machine name, followed by the port, i.e., leave out the domain name part. The live public PRIME server is available at https://rr-prime.ai.sri.com/.

2) A login window for PRIME will appear (Figure 3).



Welcome to the PRIME Server

PRIME Version: 2.0
KB Version: 7.0
Latest Patches: 20 Mar 2007
Number of Connections: 0

User Name: iharrison
Password:

TiP: Browser cookies need to be turned on in order to login to the PRIME server.

For information about SEAS, visit the SEAS Homepage at http://www.seas.sri.com
SEAS 7.1 Patent Pending and Unpublished Copyright © 1998-2007, SRI International
PRIME 2.0, Unpublished Copyright © 2006-2007, SRI International

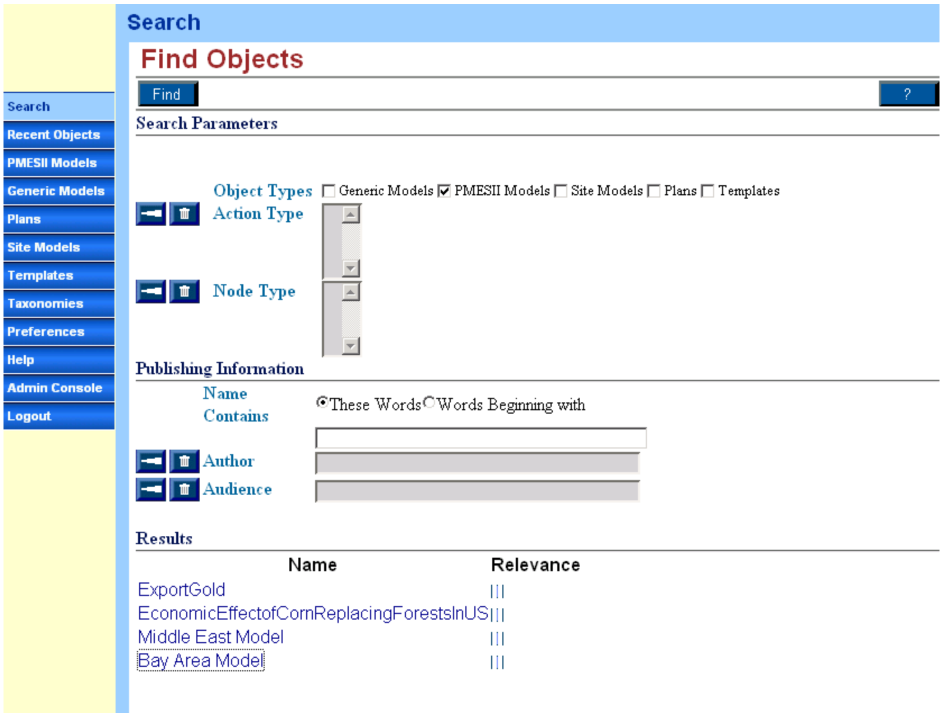
Figure 3: PRIME Login Window

4.3.2 Main PRIME Manager Window

The main PRIME manager window appears after a user has successfully logged into a PRIME server (Figure 7). This manager window is divided into two main frames. To the left is a series of tabs representing the different tasks/objects that a user is interested in when conducting PMESII modeling. To the right is displayed information pertaining to which tab has been pressed. By default, when the main manager window is opened the **Search** tab is selected and PRIME's Find dialog is displayed in the frame on the right. The functionality for the following tabs is discussed in later sections: PMESII Models (Section 4.7.1), Generic Models (Section 4.6.1), Plans (Section 4.6.3), Site Models (Section 4.6.2), Templates (Section 4.5.2), and Taxonomies (Section 4.5.1). The remaining tabs (Search, Recent Objects, Preferences, Help, and Logout) are discussed below.

Search

Clicking on the Search tab changes the frame on the right of main manager window to show the interface used to help find objects in PRIME, as shown in Figure 4. When hundreds of objects of all different types are in PRIME's database, then using a file-explorer-like interface to find objects becomes impossible, as has been SRI's experience with operational use of SEAS. Hence, the PRIME Search interface becomes the primary method for finding objects.



Search

Find Objects

Search Parameters

Object Types ☐ Generic Models ☑ PMESII Models ☐ Site Models ☐ Plans ☐ Templates

Action Type

Node Type

Publishing Information

Name Contains ● These Words ○ Words Beginning with

Author

Audience

Results

Name	Relevance
ExportGold	|||
EconomicEffectofCornReplacingForestsInUS	|||
Middle East Model	|||
Bay Area Model	|||

Figure 4: Search Window

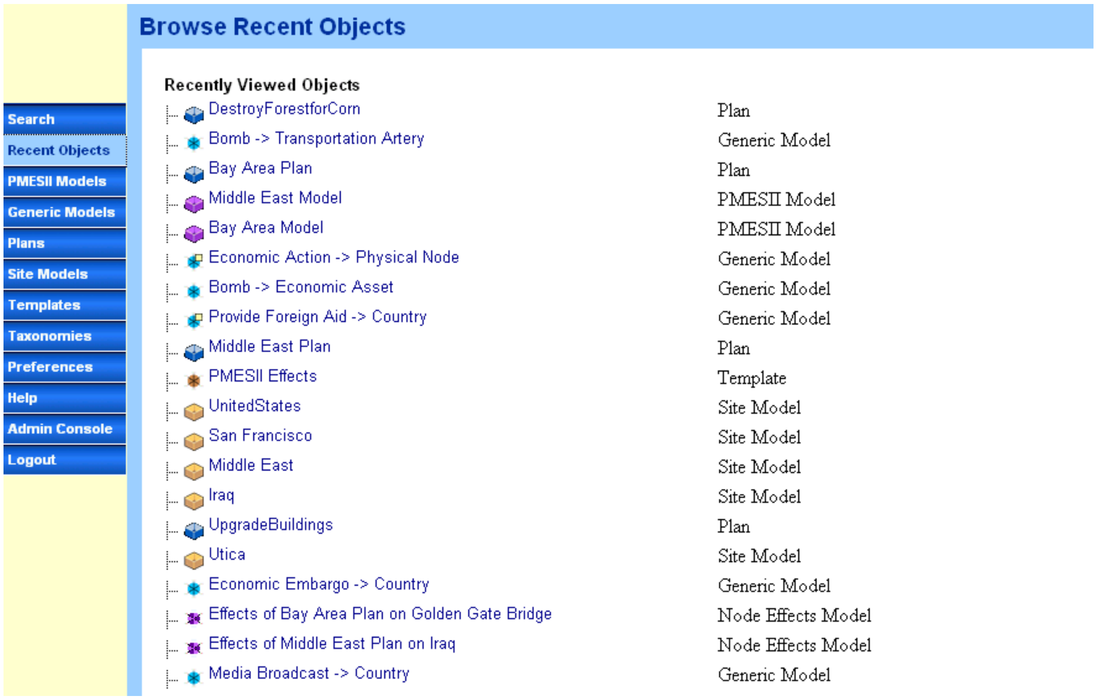
PRIME provides several different ways to search for objects that a user is interested in and ways to filter the number of results, based on knowledge of what it is the user is looking for. A user who knows the type of object can check the box corresponding to that type (one of Generic Model, PMESII Model, Site Model, Plan, or Template). A user who knows the node type or action type of the model can select it. Alternatively, the user can simply type in part of the model title or enter the name of the model's author.

Once the user has entered all known information, clicking on the *Find* button results in any relevant found objects being displayed at the bottom of the window (Figure 4). The user can then click on an object, and the relevant object editor will open up in a separate browser window and load the selected object.

Recent Objects tab

Clicking on the Recent Objects tab changes the frame in the right of the main manager window to show those PRIME objects that the user has recently viewed (Figure 5). The information is displayed in a file-explorer-like manner, with the name and type of the PRIME object displayed.

The icons to the left of the Object names reflect the type of object, e.g., a Plan, Generic Model, PMESII Model, Site Model, and Node Effects Model. Clicking on one of these object names opens up a separate browser window and displays the object.



Browse Recent Objects

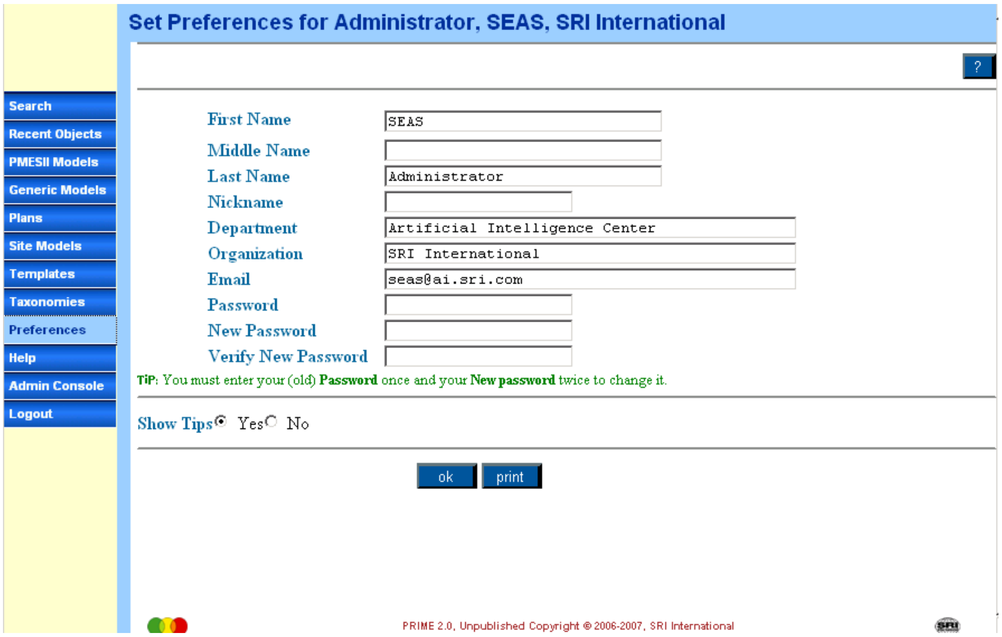
Recently Viewed Objects

DestroyForestforCorn	Plan
Bomb -> Transportation Artery	Generic Model
Bay Area Plan	Plan
Middle East Model	PMESII Model
Bay Area Model	PMESII Model
Economic Action -> Physical Node	Generic Model
Bomb -> Economic Asset	Generic Model
Provide Foreign Aid -> Country	Generic Model
Middle East Plan	Plan
PMESII Effects	Template
UnitedStates	Site Model
San Francisco	Site Model
Middle East	Site Model
Iraq	Site Model
UpgradeBuildings	Plan
Utica	Site Model
Economic Embargo -> Country	Generic Model
Effects of Bay Area Plan on Golden Gate Bridge	Node Effects Model
Effects of Middle East Plan on Iraq	Node Effects Model
Media Broadcast -> Country	Generic Model

Figure 5: Recent Objects Tab

Preferences tab

Clicking on the *Recent Objects* tab changes the main manager window right frame to show user-settable preferences (Figure 6). Currently, for PRIME these are limited to the user being able to change user profile information, change account password, and whether or not tips are to be displayed in the GUI.



Set Preferences for Administrator, SEAS, SRI International

First Name	SEAS
Middle Name	
Last Name	Administrator
Nickname	
Department	Artificial Intelligence Center
Organization	SRI International
Email	seas@ai.sri.com
Password	
New Password	
Verify New Password	

TiP: You must enter your (old) **Password** once and your **New password** twice to change it.

Show Tips ● Yes ○ No

PRIME 2.0, Unpublished Copyright © 2006-2007, SRI International

Figure 6: Preferences Tab

Help tab

Clicking on the *Help* tab opens a new browser window that loads the latest version of the PRIME user manual.

Logout tab

Clicking on the *Logout* tab logs the user out of the PRIME server and changes the browser window to display the PRIME login window (Figure 3). It should also be noted that a user will be automatically logged out after a certain period of time (default 1 hour) of inactivity. This inactivity time is server configurable, and is settable in the configuration file used as part of the PRIME server startup, as described in the PRIME Software Release Instructions documentation that is included in the PRIME software distribution. Hence, organizations can modify this inactivity timeout value to their own liking.

4.4 PRIME Installation and Initialization

4.4.1 PRIME Software Distribution

PRIME is distributed as a zip file and runs on a Sun Microsystems Solaris workstation/server, running Solaris 7 or better. A typical machine would have 500+ MB of RAM and at least 1 GB of available disk space.

The PRIME client software is in theory any modern browser, which can be running on any operating system (OS). In practice only Firefox and Mozilla browsers work correctly and we have used these to conduct testing. Internet Explorer (IE) has known issues with some of the JavaScript used in the PRIME GUI, so IE **cannot** be recommended currently as a PRIME **client** browser.

The PRIME distribution contains all source code for PRIME. Building PRIME from source code requires obtaining a copy of Franz Common Lisp (www.franz.com). Building the software source code using any other Common Lisp implementation has not been tested. In addition, PRIME makes use of some Franz specific Lisp libraries. Note, however, that the PRIME distribution contains PRIME executable binaries for Sun Solaris, eliminating the need to build from source code.

The PRIME distribution contains documentation describing how to install, configure, and start a PRIME server. In addition SRI is currently hosting a live public PRIME server at https://rr-prime.ai.sri.com/. Accounts can be obtained for this server by sending an email request to seas@ai.sri.com.

4.4.2 PRIME System Administration

To use PRIME, a user requires an account on the PRIME server. In addition to regular user accounts, each server has a super-user administrator account, with username `administrator`. An administrator user can see everything that all users have created on the server, and can also administer the PRIME server. When a user logs in to PRIME using an administrator account, the main PRIME manager window also includes an Admin Console tab on the left side (Figure 7). Other than this tab, the GUI for an administrator user is identical to the GUI for a general user.

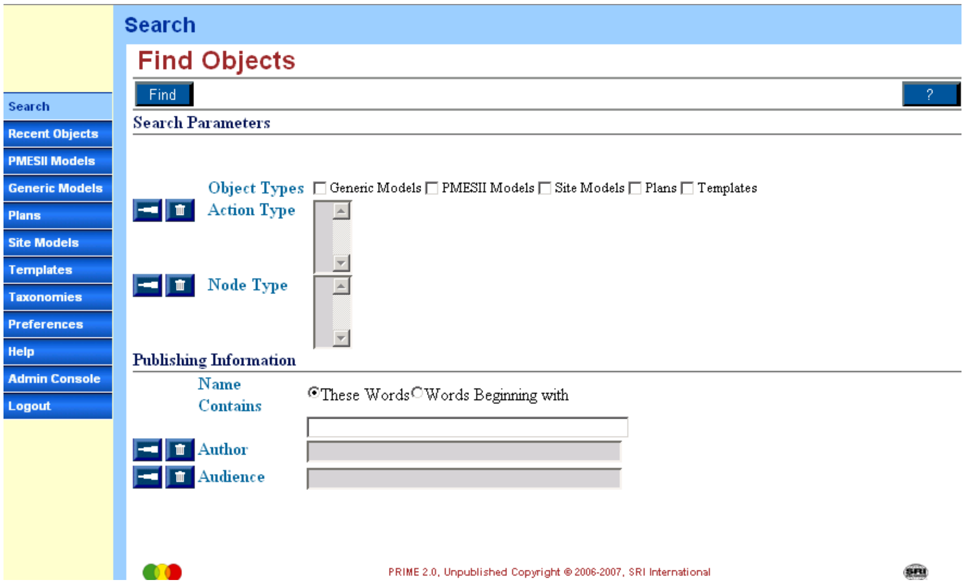


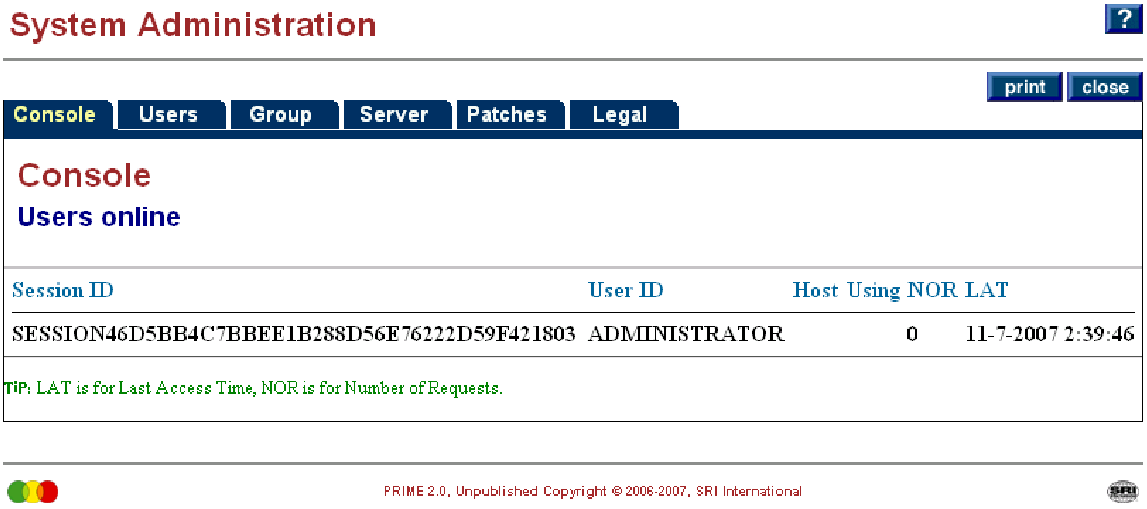
Figure 7: Main PRIME Manager Window

Clicking on the Admin Console tab on the left of the main PRIME manager window opens up the system administration window.

The System Administration GUI has six tabs: Console, Users, Groups, Server, Patches, and Legal. These are described below:

Console tab

Clicking on the *Console* tab displays which users are currently logged in to the PRIME server (Figure 8).



Console

Users online

Session ID	User ID	Host	Using	NOR	LAT
SESSION46D5BB4C7BBEE1B288D56E76222D59F421803	ADMINISTRATOR			0	11-7-2007 2:39:46

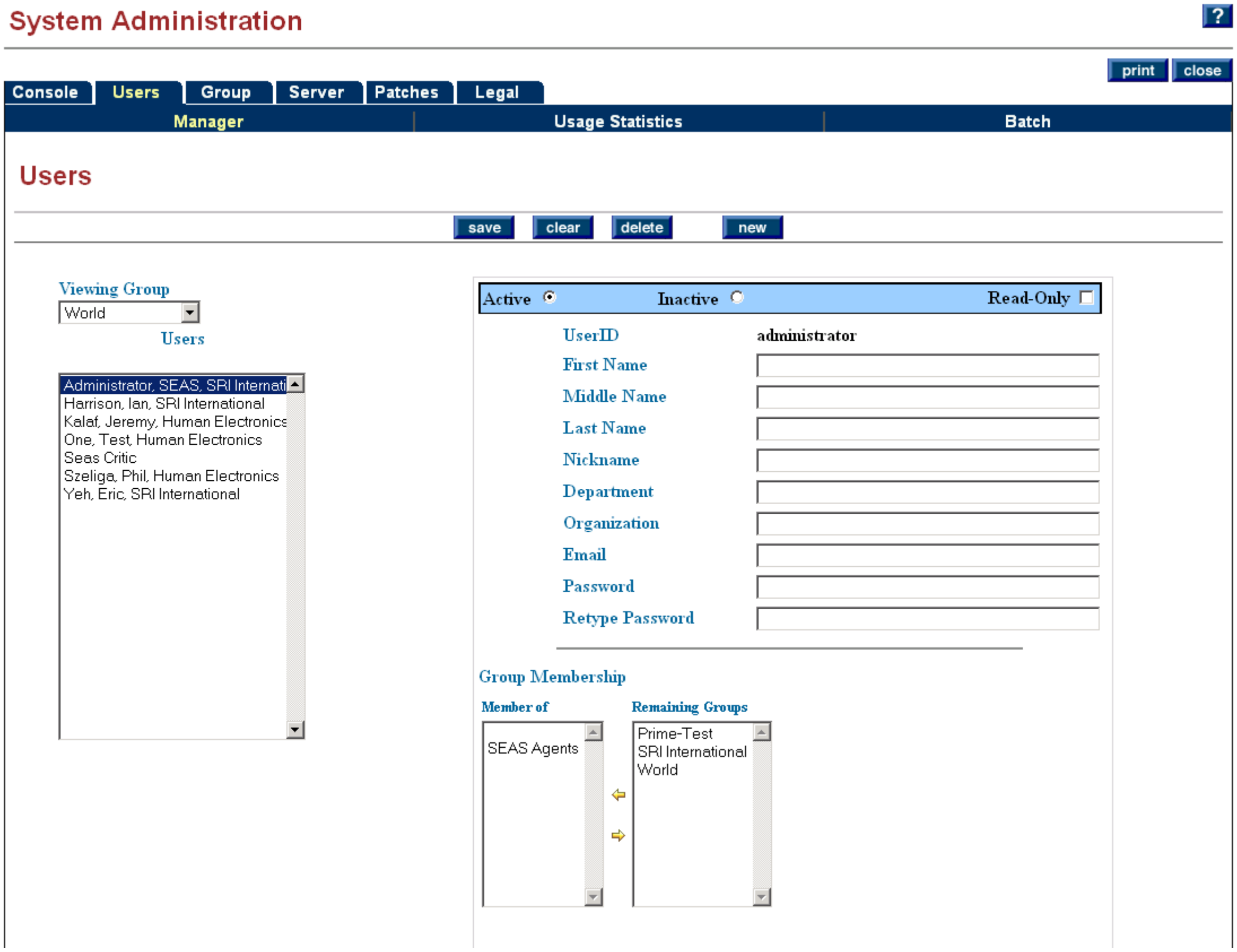
TiP: LAT is for Last Access Time, NOR is for Number of Requests.

PRIME 2.0, Unpublished Copyright © 2006-2007, SRI International

Figure 8: Console Tab

User tab

Clicking on the *User* tab displays the user administration GUI (Figure 9), which allows the administrator to add/delete/modify user account information. Each user is given a username and password for logging in to the PRIME server. Users can be assigned to a group, too, or to multiple groups. This is useful, in that objects in PRIME may be made read/write or read-only with respect to a group, rather than having to list all users. This means that a user who has a new account created and is assigned to an existing group automatically gets to see all objects available to that group.



Viewing Group: World

Users

Administrator, SEAS, SRI Internati
Harrison, Ian, SRI International
Kalaf, Jeremy, Human Electronics
One, Test, Human Electronics
Seas Critic
Szeliga, Phil, Human Electronics
Yeh, Eric, SRI International

Active ● Inactive ○ Read-Only □

UserID	administrator
First Name	
Middle Name	
Last Name	
Nickname	
Department	
Organization	
Email	
Password	
Retype Password	

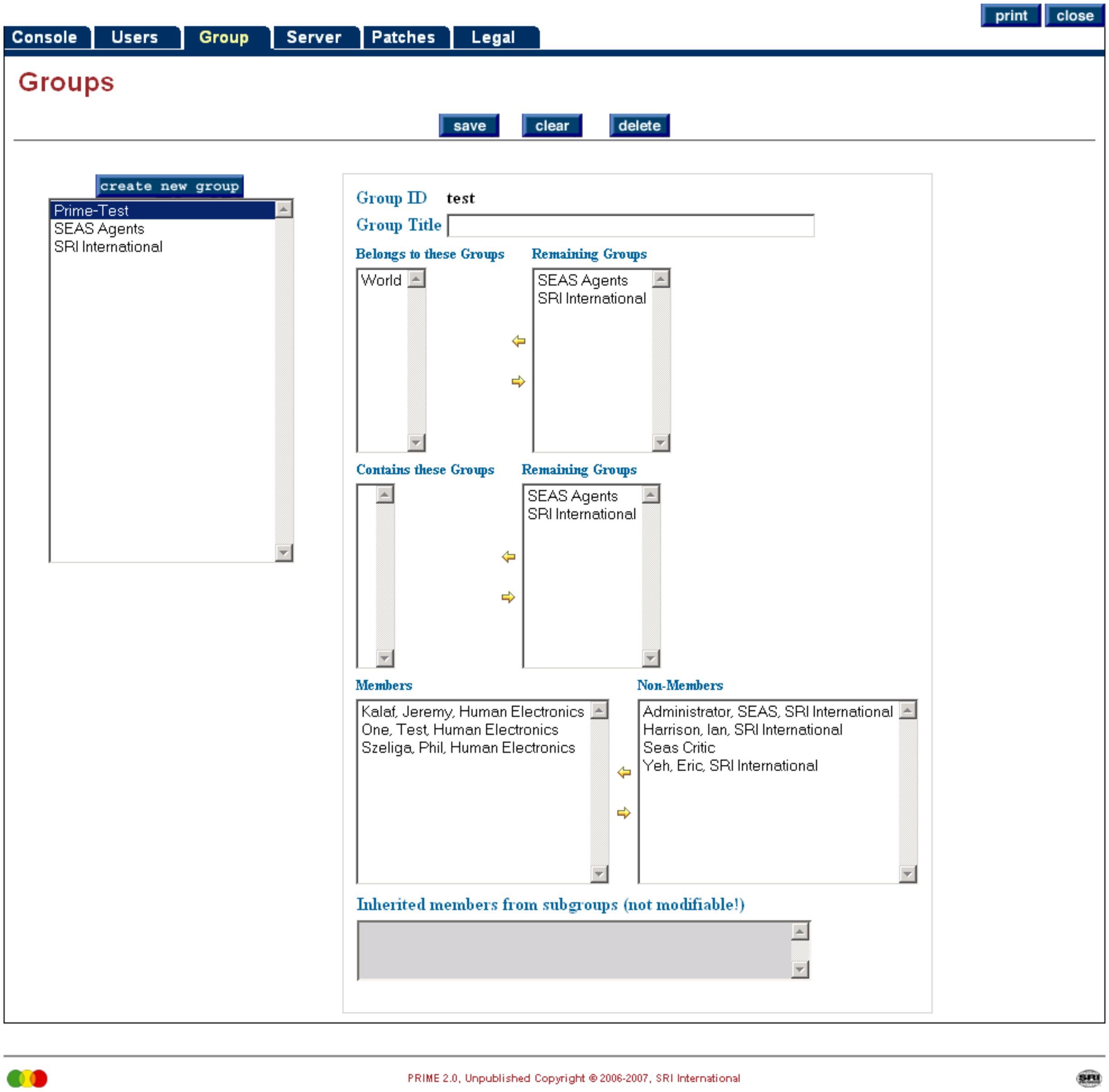
Group Membership

Member of	Remaining Groups
SEAS Agents	Prime-Test SRI International World

Figure 9: User Administration GUI

Groups tab

Clicking on the *Group* tab displays the group administration GUI (Figure 10), which allows an administrator to create/delete/modify groups and their user membership.



Prime-Test
SEAS Agents
SRI International

Group ID test

Group Title

Belongs to these Groups: World

Remaining Groups: SEAS Agents, SRI International

Contains these Groups:

Remaining Groups: SEAS Agents, SRI International

Members:
Kalaf, Jeremy, Human Electronics
One, Test, Human Electronics
Szeliga, Phil, Human Electronics

Non-Members:
Administrator, SEAS, SRI International
Harrison, Ian, SRI International
Seas Critic
Yeh, Eric, SRI International

Inherited members from subgroups (not modifiable!)

PRIME 2.0, Unpublished Copyright © 2006-2007, SRI International

Figure 10: Group Administration GUI

Server tab

Clicking on the *Server* tab displays functions for managing the PRIME server. These capabilities have been inherited from SEAS and many are not pertinent to PRIME. It is recommended that these functions are not exercised.

Patches tab
Clicking on the *Patches* tab displays functions for loading patches into the PRIME server. This allows the PRIME server to be patched without bringing the server down. Patches would be distributed by SRI.

Legal tab

Clicking on the *Legal* tab displays legal information about the copyright and ownership of all code components used to create the PRIME software.

4.5 Organizational Tailoring of PRIME

With PRIME installed and user accounts established, the next task for an organization using PRIME is to tailor PRIME to an organization-specific approach to PMESII modeling and its specialized domain terminology. Tailoring PRIME consists of two main things: the development of the PMESII node/action taxonomy that is to be used for PRIME, and the development of the underlying PMESII model template to be used for all PMESII models. These two tasks are discussed below.

4.5.1 Creating and Editing a PMESII Node/Action Taxonomy

All PMESII effect models in PRIME capture the effect of action(s) on node(s). Rather than just capturing the name of the action or node, PRIME requires that the class (type) of each node and action must be captured, as this is important meta-data. PRIME contains a node/action taxonomy editor allowing users to define and edit the node/action taxonomy that they intend to use for their PMESII modeling.

The normal CONOPS for PRIME node/action taxonomy development is that it is undertaken by an organization prior to PMESII effects model development. Taxonomies will naturally be extended, where it is found that detail is insufficient for a particular modeling task, but it is important to get the fundamentals of the taxonomies correct before doing extensive PMESII model development.

The PRIME taxonomy editor is accessed from the main PRIME manger window by clicking on the Taxonomies tab. The right frame displays the taxonomy editor/viewer (Figure 11). PRIME adopts the CPE program core taxonomy (developed by ISX and available to all CPE participants) for its baseline terms, such as Node and Action, and all subsequent terms are sub-classed off these root terms.

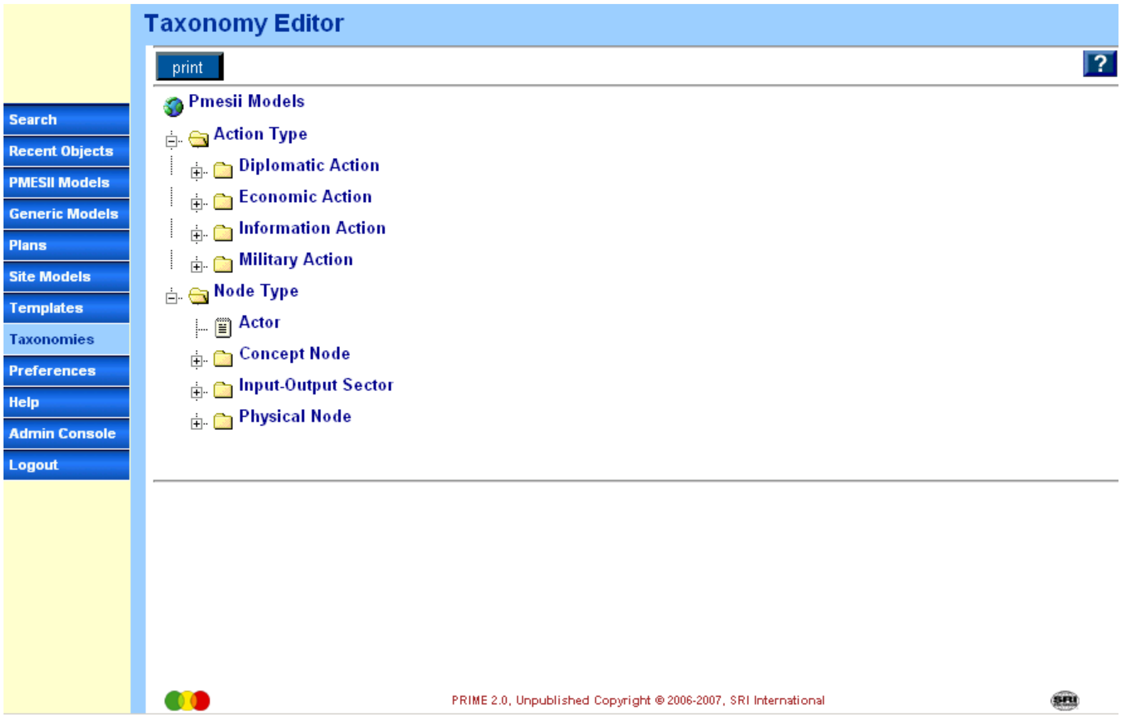


Figure 11: PRIME Taxonomy Editor/Viewer

The action and node type taxonomies are displayed in a file-explorer-like tree. A user can add an action or node type by selecting a type to be the parent type of the new type. To the left of the parent type appear two buttons, one of which is an add button. Clicking on the add button brings up a dialog box where the user types in the new child type. In addition, a user can rename a type, for example, to correct a spelling mistake, by clicking on the type and then clicking on the rename button that then appears to the left of the selected type.

Note that taxonomy deletes are not possible because all PMESII models, and Plans and Site Models, use the loaded taxonomy. If a taxonomic type was deleted and this type was used in existing models, then this would invalidate the existing models. It was decided that the task of allowing a taxonomic delete would take considerable project resources, which would outweigh the benefits for the project. Hence taxonomic management and tracking has been left as future research.

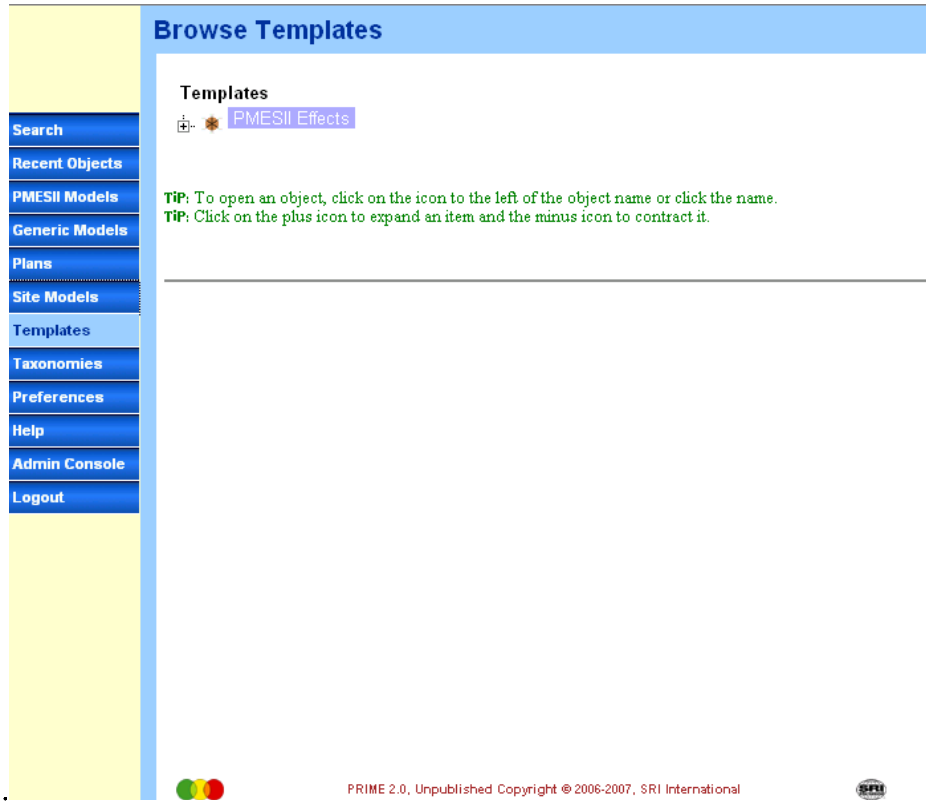
4.5.2 Creating and Editing a PMESII Effects Model Template

All PMESII effects models in PRIME are based on a PMESII effects model *template.* A PRIME template defines the analytic structure and methods for the model. For PRIME PMESII effects models, there is a single over-arching template – the PMESII direct-effects template. This template has six independent dimensions, related to the six dimensions of a PMESII analysis: political, military, economic, social, infrastructural, and informational.

The PRIME software comes with a default direct effects PMESII template, but organizations can adapt this default template to suit their own purposes.

As with taxonomy development, PMESII model template development is likely to be undertaken by an organization prior to developing a library of generic models and any crisis-specific models. Template design is often conducted in a brainstorming, both top-down and bottom-up, manner. Example models often are developed based on previous known cases, in order to test the goodness of a template. Organizations typically cycle though several iterations of template creation, using subsequent model creation as a validation process for the suitability, coverage, and expressiveness of the template. At this point the template design has solidified and is frozen until being used in an operational setting.

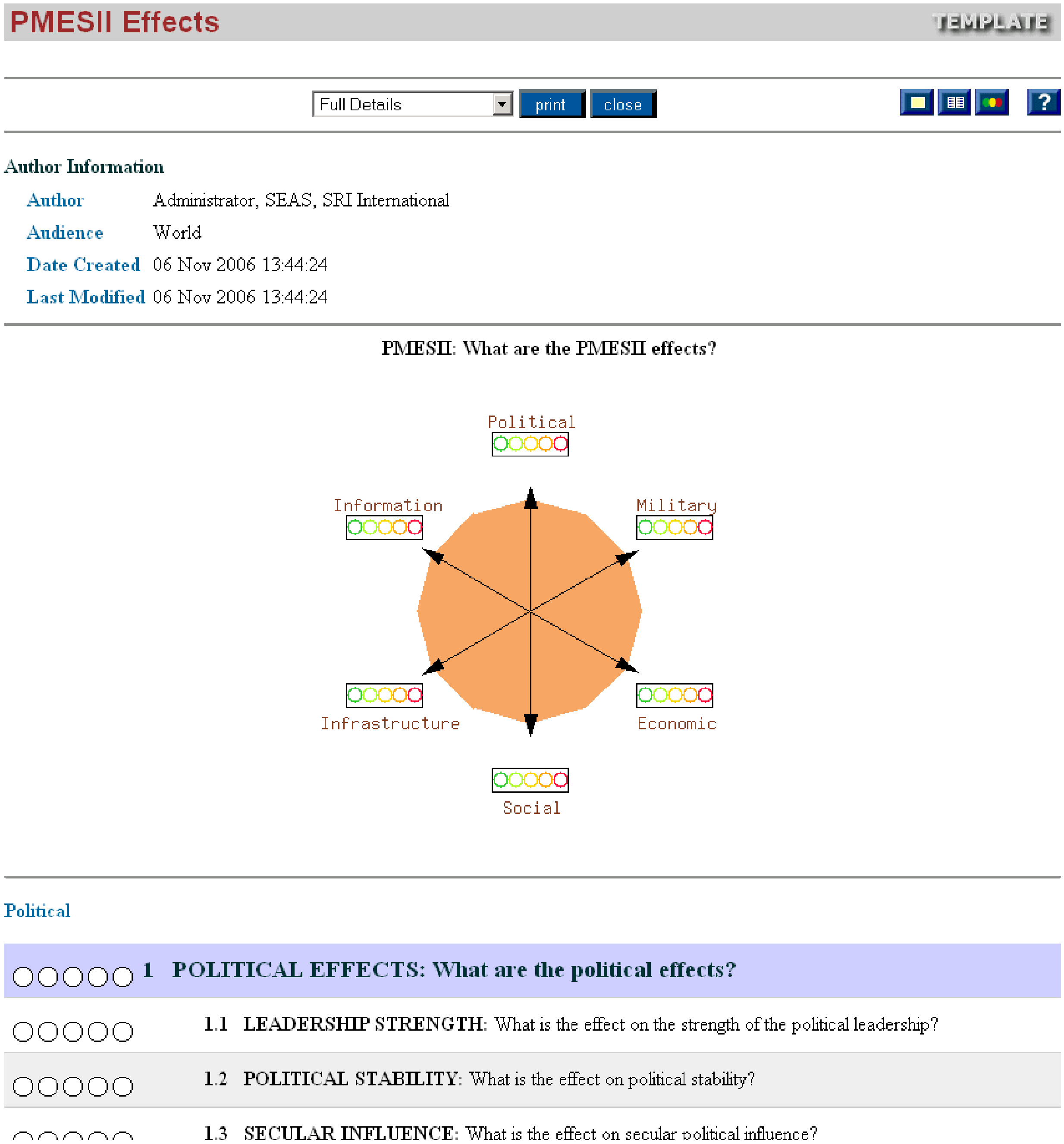
The PRIME template editor is accessed from the main PRIME manager window by clicking on the *Templates* tab. The right frame displays the single PMESII model template (Figure 12). Clicking on this template opens up the template editor in a separate browser and loads the selected template (Figure 13).



TiP: To open an object, click on the icon to the left of the object name or click the name.
TiP: Click on the plus icon to expand an item and the minus icon to contract it.

PRIME 2.0, Unpublished Copyright © 2006-2007, SRI International

Figure 12: Main Window with *Template* Tab Selected



Author Information

Author	Administrator, SEAS, SRI International
Audience	World
Date Created	06 Nov 2006 13:44:24
Last Modified	06 Nov 2006 13:44:24

PMESII: What are the PMESII effects?

Political

○○○○○	1	**POLITICAL EFFECTS: What are the political effects?**
○○○○○	1.1	**LEADERSHIP STRENGTH:** What is the effect on the strength of the political leadership?
○○○○○	1.2	**POLITICAL STABILITY:** What is the effect on political stability?
○○○○○	1.3	**SECULAR INFLUENCE:** What is the effect on secular political influence?

Figure 13: Template Editor

The template has the six PMESII dimensions, and for each dimension there is a root question to be answered. For the political dimension, for example, the root question is, “What are the political effects?” To answer this question, we can break the effects question down into sub-questions that address sub-effects. These dimensional root questions and branch sub-questions collectively constitute the PMESII model template. The template will subsequently be applied to create Generic PMESII model situations where an action is being taken on something, so that the question is then, for example, “What is the political effect of applying a trade embargo on a country?” However, the template itself is situation neutral.

PRIME inherits much of the template editing functionality from SEAS. In SEAS, templates can be arbitrarily wide and deep, although common practice is to keep templates as narrow and shallow as possible, while still being of practical use. For PRIME, we decided to make templates a fixed size – only one branch deep and with only six sub-questions. This decision means that all templates (and hence models derived from the templates), would be identical in structure. This has large benefits for PRIME, as it allows for easy model merging and comparison. However, it is acknowledged that this is restrictive and that some organizations may want to have a different template structure than what is supplied.

Template Editor Functionality

Below the title of the template in Figure 13 is the toolbar and directly below the toolbar is basic information about the template (who can edit it, who can view it, date created, date last modified). The toolbar provides the following capabilities.

Detail Setting: The user can set how much detail to view in the editor from one of three preset levels (Full Details, Moderate Details, and Minimal Details).

Print: The browser window Print dialog opens, allowing the user to print out the template.

Close: The Template Editor closes. Remember, PRIME is not like a Windows application: there are no save buttons anywhere. Edits are made instantaneously and cannot be undone.

Memos: Clicking on the ▢ button, opens up the PRIME memo dialog, which allows users to leave post-it-like notes for collaborators. Memos can be left on all the other object editors in PRIME (Site model, Plan, Generic model, PMESII model).

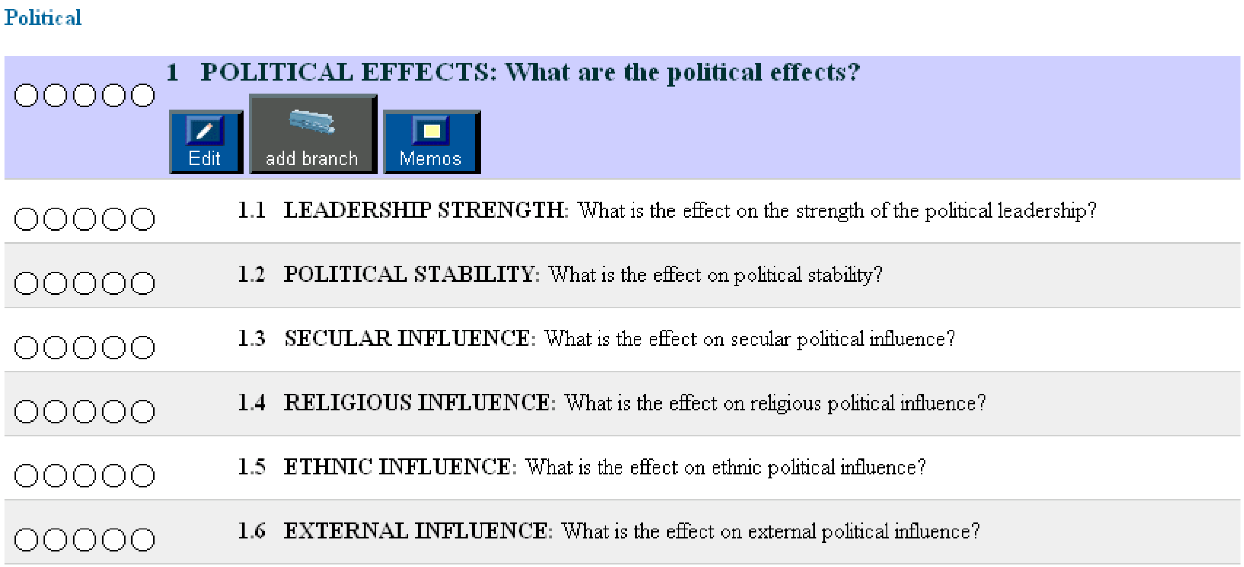
Access Control: Clicking on the ▤ button opens up the PRIME publication information dialog box. All objects in PRIME (Site models, Plans, Generic models, PMESII models) have associated access control. This information says who can edit the object (an **author** in PRIME terminology) and who can just view (an **audience** member in PRIME terminology

Main Window: Clicking on the ●●● button brings the main PRIME manager window to the front. With many windows open the manager window can often get buried, and clicking on this button exposes it.

Help: Clicking on the ? button brings up PRIME's help system in a separate browser window.

The rest of the template editor displays the actual template – first as a high-level graphic depicting the six dimensions of the PMESII model, and then as a tree-like breakdown of the six PMESII dimensions. Clicking on a root dimension question brings up the in-line editing capabilities for a root question (Figure 14). By clicking on the *Edit* button, the user can edit the root question text. The add branch button is currently disabled in PRIME. Clicking on the *Memos* button brings up the PRIME memo dialog, allowing a user to leave a memo for template development collaborators. An added memo is displayed inline for any user who received the memo (Figure 15). In the memo dialog the user can choose to un-display the memo, edit it (if that user wrote it), or delete it (if that user wrote it).

With the root dimension question selected, if the user clicks on another root question or any other sub-question, the displayed root question editing toolbar goes away.



Political

○○○○○ 1 **POLITICAL EFFECTS: What are the political effects?**

○○○○○ 1.1 **LEADERSHIP STRENGTH**: What is the effect on the strength of the political leadership?

○○○○○ 1.2 **POLITICAL STABILITY**: What is the effect on political stability?

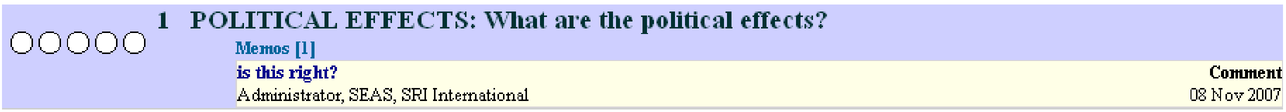
○○○○○ 1.3 **SECULAR INFLUENCE**: What is the effect on secular political influence?

○○○○○ 1.4 **RELIGIOUS INFLUENCE**: What is the effect on religious political influence?

○○○○○ 1.5 **ETHNIC INFLUENCE**: What is the effect on ethnic political influence?

○○○○○ 1.6 **EXTERNAL INFLUENCE**: What is the effect on external political influence?

Figure 14: Editing Root Dimension Question



○○○○○ 1 **POLITICAL EFFECTS: What are the political effects?**

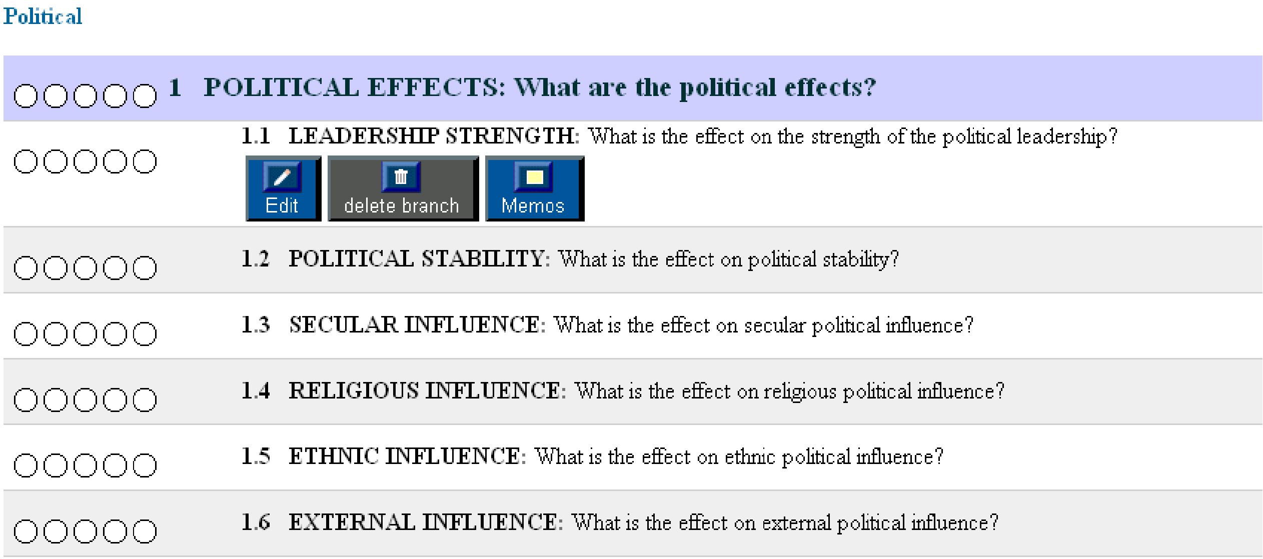
Memos [1]

is this right? **Comment**

Administrator, SEAS, SRI International 08 Nov 2007

Figure 15: Inline Memos Displayed

When the user clicks on a sub-question, the inline sub-question toolbar appears (Figure 16). This toolbar allows the user to edit the question text and add a memo. Currently, the ability to delete the sub-question itself is disabled.



Political

○○○○○ 1 POLITICAL EFFECTS: What are the political effects?

○○○○○ 1.1 LEADERSHIP STRENGTH: What is the effect on the strength of the political leadership?

○○○○○ 1.2 POLITICAL STABILITY: What is the effect on political stability?

○○○○○ 1.3 SECULAR INFLUENCE: What is the effect on secular political influence?

○○○○○ 1.4 RELIGIOUS INFLUENCE: What is the effect on religious political influence?

○○○○○ 1.5 ETHNIC INFLUENCE: What is the effect on ethnic political influence?

○○○○○ 1.6 EXTERNAL INFLUENCE: What is the effect on external political influence?

Figure 16: Editing Sub-question

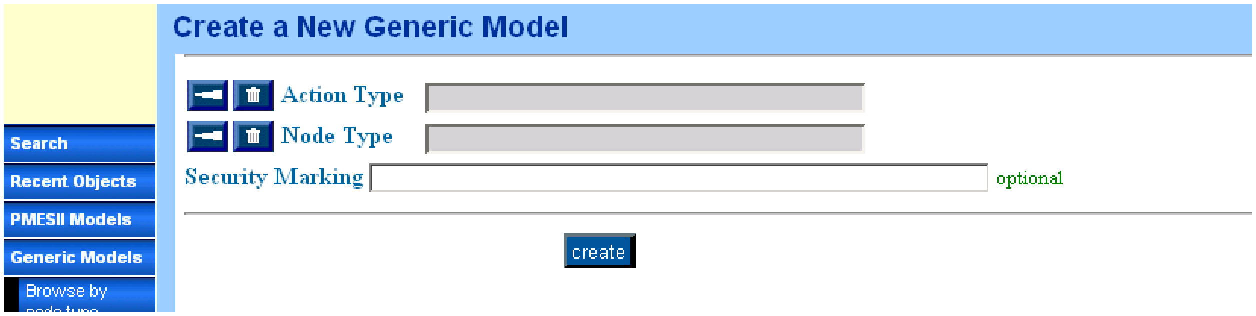
4.6 Pre-crisis Use of PRIME

Once PRIME has been tailored for use within an organization (or the default distribution template and node/action taxonomies are chosen as being acceptable), it can be used to develop pre-crisis models and plans. All the models and plans discussed below may actually be produced during a crisis, but from the identified use cases, many of these would conventionally be done in pre-crisis mode.

4.6.1 Creating a Generic PMESII Effects Model

From a PRIME CONOPS perspective the major pre-crisis use of PRIME is to develop a library of generic PMESII effects models for a range of relevant node type/action types. PRIME comes with a set of example generic models for two scenarios: affecting change on a country, and bombing a node. The example of affecting change on a country is used below. Note that all generic models consider the direct effects of a single action type on a single node type – for example, the effect of an economic embargo (action type) on a country (node type).

The PRIME generic models are accessed from the main PRIME manager window by clicking on the **Generic Models** tab. The tab expands to present three sub-options: browsing by node type, browsing by action type, and creating a new model. Clicking on *browse by action* or *browse by node type* displays a file-explorer-like tree hierarchy of either the action type or node type taxonomies that have been defined for use with PRIME (see Section 4.5.1). Expanding the branches on this taxonomy leads to finding Generic Models that have that action or node type associated with them. Clicking on the *Create a New Model* tab brings up, in the right frame of the main PRIME manager window, the new generic model dialog (Figure 17). The user clicks on the ▬ button to bring up the node or action type taxonomy browser (Figure 18).



Create a New Generic Model

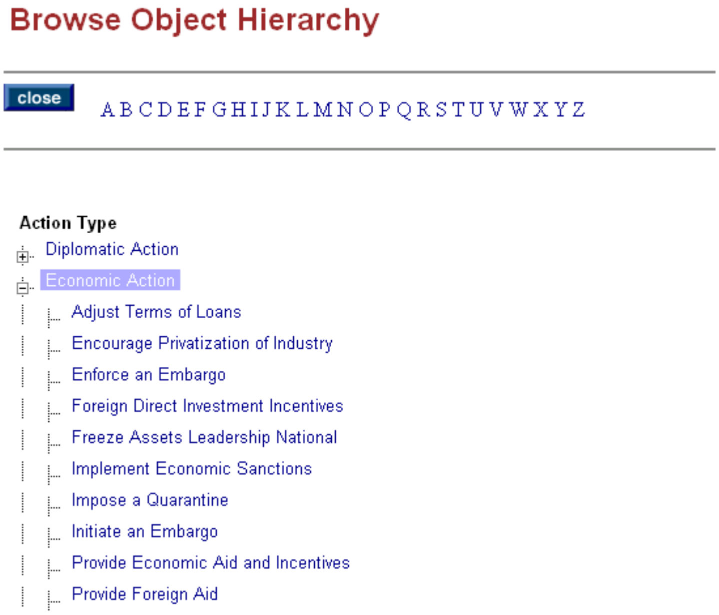
Action Type

Node Type

Security Marking optional

create

Figure 17: Creating a New Generic Model



Browse Object Hierarchy

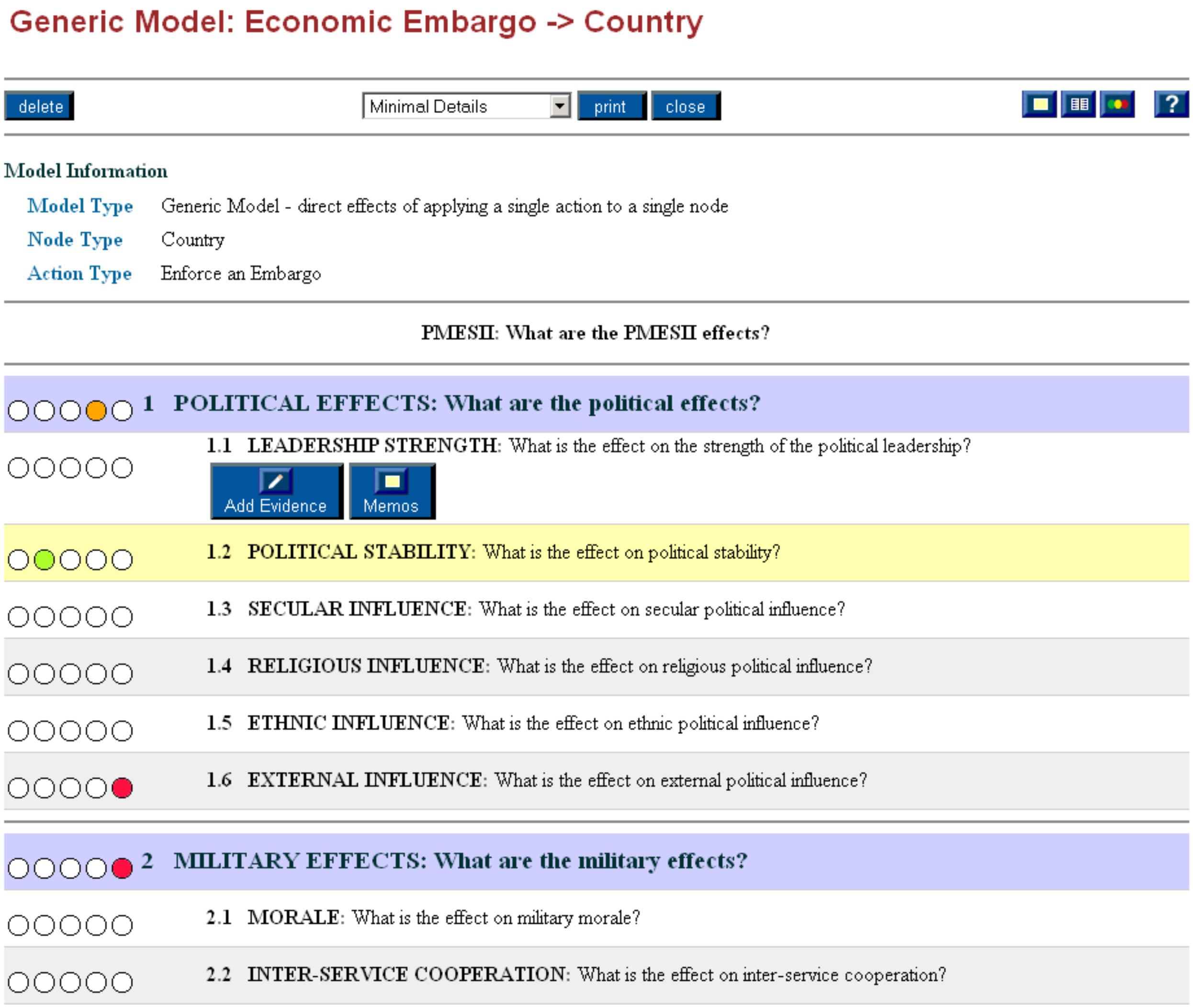
close A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Action Type

- Diplomatic Action
- Economic Action
  - Adjust Terms of Loans
  - Encourage Privatization of Industry
  - Enforce an Embargo
  - Foreign Direct Investment Incentives
  - Freeze Assets Leadership National
  - Implement Economic Sanctions
  - Impose a Quarantine
  - Initiate an Embargo
  - Provide Economic Aid and Incentives
  - Provide Foreign Aid

Figure 18: Taxonomy Browser

The user navigates through the taxonomy hierarchy and selects a node type and action type for the generic model. The user then clicks on the create button, and the generic model editor window appears (Figure 19) with an empty generic model. The structure of this generic model comes from PRIME’s PMESII model template (see Section 4.5.2).



Generic Model: Economic Embargo -> Country

Model Information

Model Type: Generic Model - direct effects of applying a single action to a single node

Node Type: Country

Action Type: Enforce an Embargo

PMESII: What are the PMESII effects?

○○○●○ 1 POLITICAL EFFECTS: What are the political effects?

○○○○○ 1.1 LEADERSHIP STRENGTH: What is the effect on the strength of the political leadership?

○●○○○ 1.2 POLITICAL STABILITY: What is the effect on political stability?

○○○○○ 1.3 SECULAR INFLUENCE: What is the effect on secular political influence?

○○○○○ 1.4 RELIGIOUS INFLUENCE: What is the effect on religious political influence?

○○○○○ 1.5 ETHNIC INFLUENCE: What is the effect on ethnic political influence?

○○○○● 1.6 EXTERNAL INFLUENCE: What is the effect on external political influence?

○○○○● 2 MILITARY EFFECTS: What are the military effects?

○○○○○ 2.1 MORALE: What is the effect on military morale?

○○○○○ 2.2 INTER-SERVICE COOPERATION: What is the effect on inter-service cooperation?

Figure 19: Generic Model Editor

The Generic Model editor has the following functionality in its toolbar:

Delete: Lets a user with write access to the generic model delete it

Delete: Prints the generic model

Close: Closes the generic model editor window

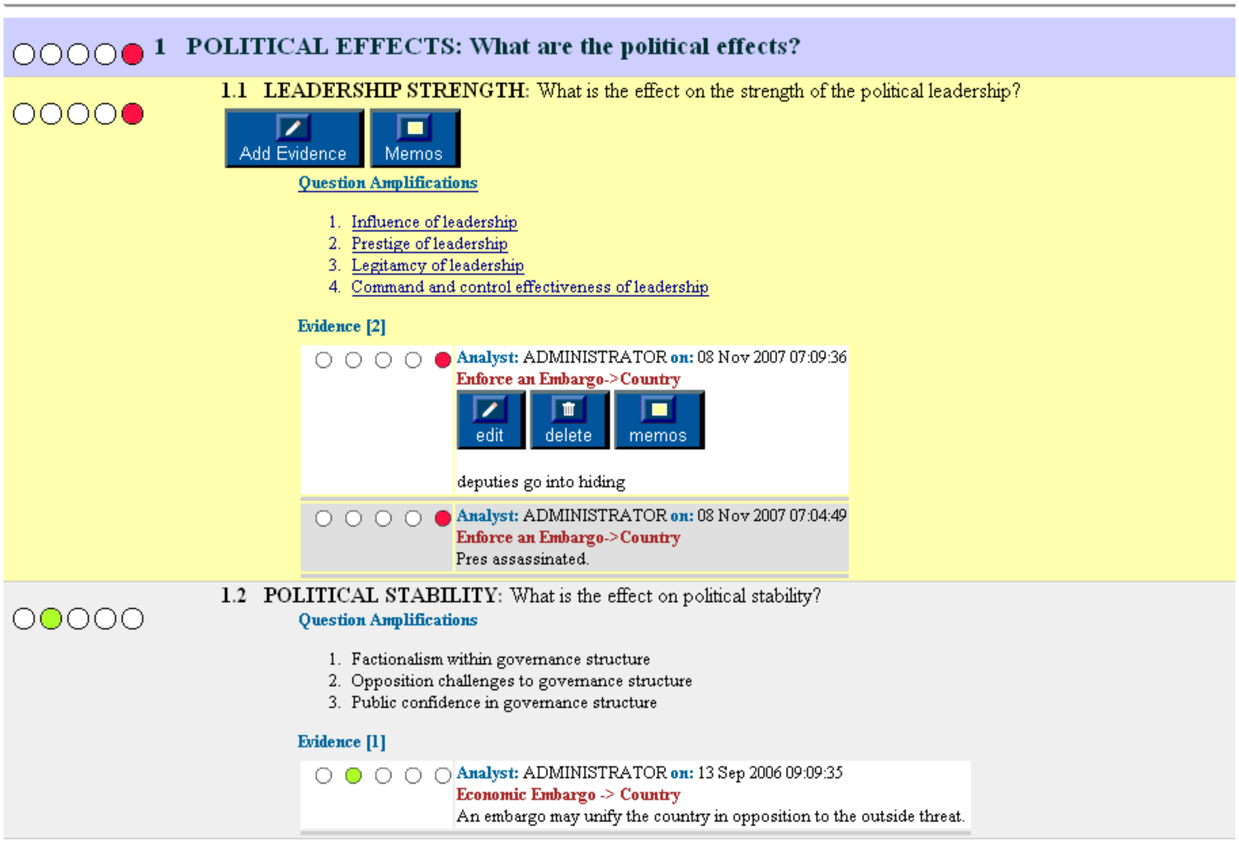
▣**:** Opens memo dialog

▤: Opens access control dialog

●: Brings main PRIME window to foreground

?: Displays help

Below the toolbar is some meta-data about the model itself, defining the action and node type of the generic model. Below this is the heart of the generic model editor (Figure 20).



○○○○● 1 POLITICAL EFFECTS: What are the political effects?

○○○○● 1.1 LEADERSHIP STRENGTH: What is the effect on the strength of the political leadership?

Question Amplifications

1. Influence of leadership
2. Prestige of leadership
3. Legitamcy of leadership
4. Command and control effectiveness of leadership

Evidence [2]

○ ○ ○ ○ ● Analyst: ADMINISTRATOR on: 08 Nov 2007 07:09:36
Enforce an Embargo->Country
deputies go into hiding

○ ○ ○ ○ ● Analyst: ADMINISTRATOR on: 08 Nov 2007 07:04:49
Enforce an Embargo->Country
Pres assassinated.

○●○○○ 1.2 POLITICAL STABILITY: What is the effect on political stability?

Question Amplifications

1. Factionalism within governance structure
2. Opposition challenges to governance structure
3. Public confidence in governance structure

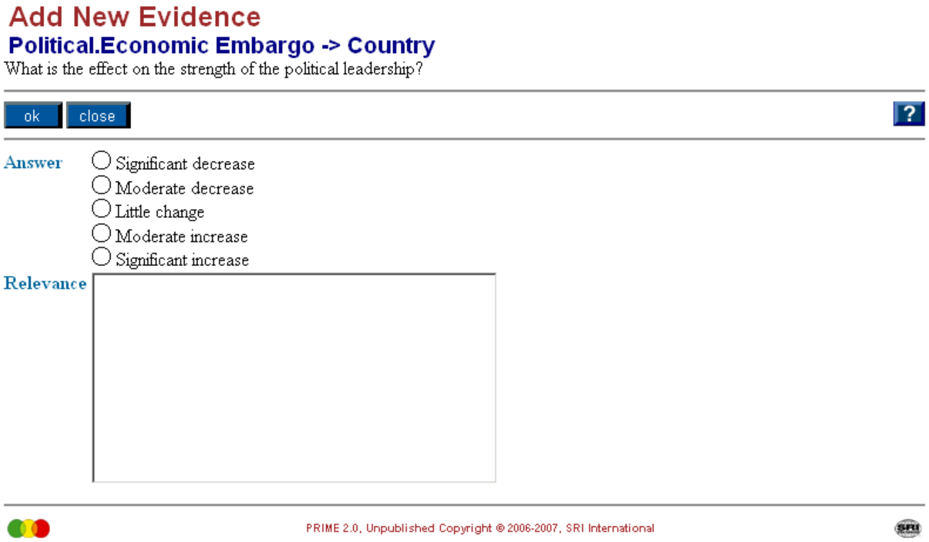
Evidence [1]

○ ● ○ ○ ○ Analyst: ADMINISTRATOR on: 13 Sep 2006 09:09:35
Economic Embargo -> Country
An embargo may unify the country in opposition to the outside threat.

Figure 20: Generic Model Editor Detail

The user goal is to then examine all the questions for each of the six PMESII dimensions defined in the PMESII model template, and consider the question of what are the specific predicted effects of the stated action type on the stated node type. For instance, in Figure 20 the user is asked to specify what the predicted effect on the strength of the political leadership is if an economic embargo is applied on a country.

To do this, a user states the predicted effects by clicking on the **Add New Evidence** button. This opens up the Evidence Editor (Figure 21).


Add New Evidence

Political.Economic Embargo -> Country

What is the effect on the strength of the political leadership?

Answer
○ Significant decrease
○ Moderate decrease
○ Little change
○ Moderate increase
○ Significant increase

Relevance

PRIME 2.0, Unpublished Copyright © 2006-2007, SRI International

Figure 21: Evidence Editor

The user is expected to provide two basic kinds of information: an answer to the question by selecting a choice from one of the set answers given, and some text as to why the user has determined that the effect is as stated. The user provides answers and evidence to all relevant questions in all six dimensions of the generic model. In general an action on a node will affect only a few questions on a few dimensions. It is up to the analyst to examine each of the generic model questions and determine whether the action on the node has any effect. Answers do not have to be just a single prediction. They may be multiple contiguous answers representing a range of belief/uncertainty in their answer/prediction.

When the user clicks on the *OK* button in the Evidence Editor the window closes and the evidence is added to the generic model. Figure 20 shows an example where two pieces of evidence have been provided by the user for a single sub-question. For each piece of evidence it is shown who added the evidence, when the evidence was added, the answer to the question, and some text to back up the answer. Users can provide as many pieces of evidence as they wish. A user who has write access for the generic model can edit or delete any evidence and can leave a memo for another collaborator. Where two pieces of evidence are given, the two answers are combined using a combining algorithm called Bounds, which is one of SEAS's built-in data fusion algorithms and is defined as

> **Bounds:** An automated fusion method that returns all answers between and including the lowest- and highest-valued answers.

For PMESII modeling it was decided that showing the bounds of the spread in the uncertainty of the effect of an action type on a node type was important. Hence, if two analysts both viewed an action type on a node type as having the effect of significant decrease of some measure, then since they were in agreement the combined result would be the same – a significant decrease. However, if two analysts (or even the same analyst) enter two different answers (say significant increase to little change) then the resultant combined answer would range between these two. In this way it is communicated that there is some uncertainty about the predicted effect and the range of the uncertainty is bounded.

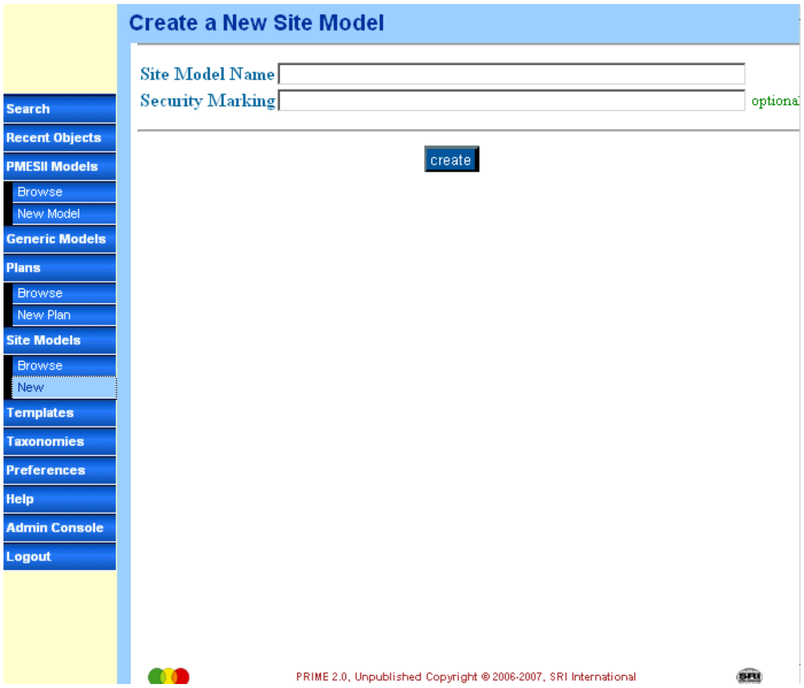
In normal use several analysts may be collaboratively developing a generic model over time, and they would use the memo facility to send memos to one another to seek clarification about answers that others had entered. The end result is that finally a complete generic model is developed that predicts the effects (on the six PMESII dimensions) that a certain action type would have on a node type.

It is anticipated that organizations would develop many of these generic models. They essentially map out the combined knowledge of a group of analysts on a number of future crisis situations. This library of generic models becomes a valuable resource for the organization, acting as a corporate memory. The models realize their value when, in a crisis situation, analysts can draw upon them to determine the predicted effects of a set of planned actions on a site (Section 4.7.1 Creating and Editing a Specific PMESII Effects Model.

4.6.2 Creating a Site Model

A site model is a collection of things (*nodes* in PMESII modeling terminology) for which the user wants to explore the effect of actions. A node may for instance be a place, piece of infrastructure, organization, or business in a city, country, or region.

The PRIME site model editor is accessed from the main PRIME manager window by clicking on the **Site Model** tab. This tab expands to allow the user to either browse all site models to which the user has access (these are displayed in the right frame of the main manager window) or to create a new site model. A user who chooses to create a new site model enters the name of the site model and then presses the create button (Figure 22). Otherwise, an existing site model can be opened by selection from the list (Figure 23). In either case, the site model editor is then displayed (Figure 24).



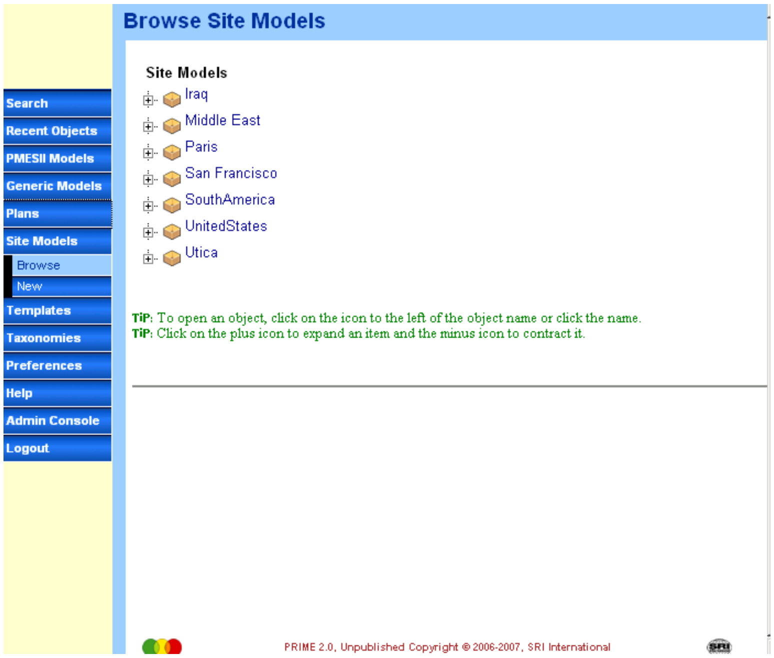
Create a New Site Model

Site Model Name

Security Marking optional

PRIME 2.0, Unpublished Copyright © 2006-2007, SRI International

Figure 22: New Site Model Creation



Browse Site Models

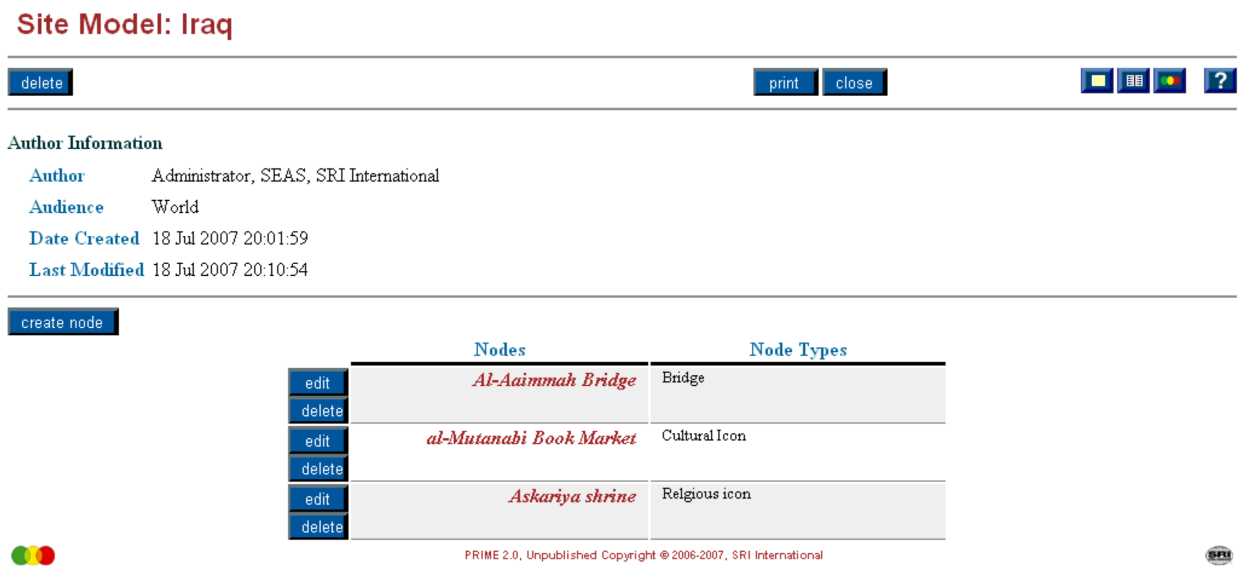
Site Models

- Iraq
- Middle East
- Paris
- San Francisco
- SouthAmerica
- UnitedStates
- Utica

TiP: To open an object, click on the icon to the left of the object name or click the name.
TiP: Click on the plus icon to expand an item and the minus icon to contract it.

PRIME 2.0, Unpublished Copyright © 2006-2007, SRI International

Figure 23: Site Model Browser



Site Model: Iraq

delete print close ▭ ▤ ●●● ?

Author Information

Author	Administrator, SEAS, SRI International
Audience	World
Date Created	18 Jul 2007 20:01:59
Last Modified	18 Jul 2007 20:10:54

create node

	Nodes	Node Types
edit delete	*Al-Aaimmah Bridge*	Bridge
edit delete	*al-Mutanabi Book Market*	Cultural Icon
edit delete	*Askariya shrine*	Relgious icon

PRIME 2.0, Unpublished Copyright © 2006-2007, SRI International

Figure 24: Site Model Editor

The site model editor has the following functionality in its toolbar:

Delete: Lets a user with write access to the site model delete it.

Delete: Prints the site model

Close: Closes the site model editor window

▭**:** Opens memo dialog

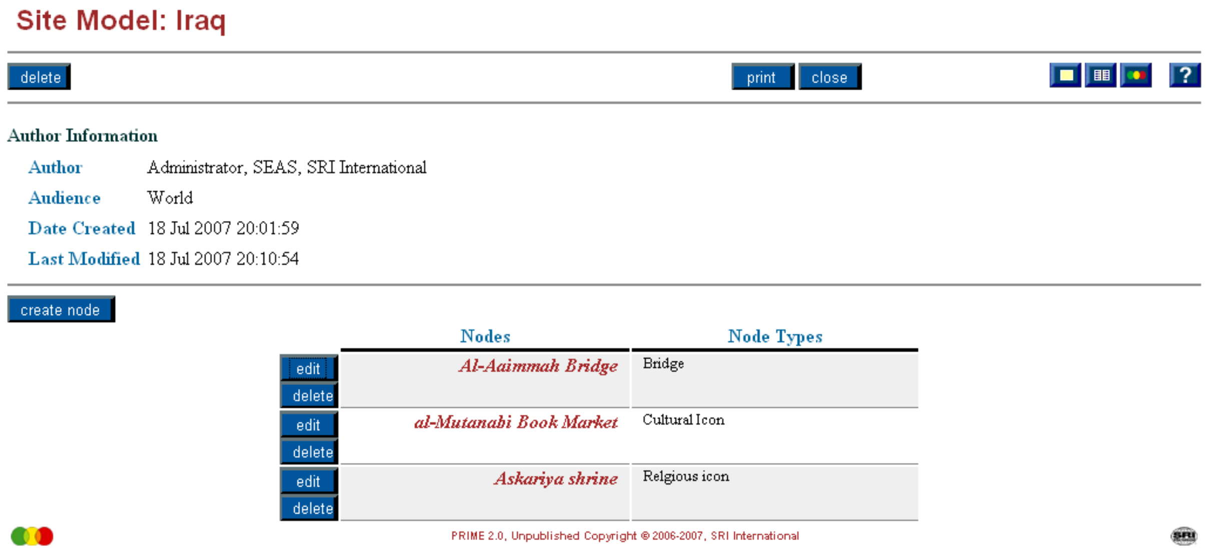
▤: Opens access control dialog

●●●: Brings main PRIME window to foreground

?: Displays help

Below the toolbar is some meta-data about the model itself (access control, date of creation, last date of modification) and below this is the actual model editor.

A user can edit or delete nodes already in a site model. Editing a node brings up the node editor dialog, which is also brought up when the user clicks on the create node button. In the node editor dialog (Figure 25) the user gives the name of the node and, most important, the type of the node. The node type comes from the node type taxonomy within PRIME (see Section 4.5.1).



Site Model: Iraq

Author Information

Author	Administrator, SEAS, SRI International
Audience	World
Date Created	18 Jul 2007 20:01:59
Last Modified	18 Jul 2007 20:10:54

Nodes	Node Types
Al-Aaimmah Bridge	Bridge
al-Mutanabi Book Market	Cultural Icon
Askariya shrine	Relgious icon

PRIME 2.0, Unpublished Copyright © 2006-2007, SRI International

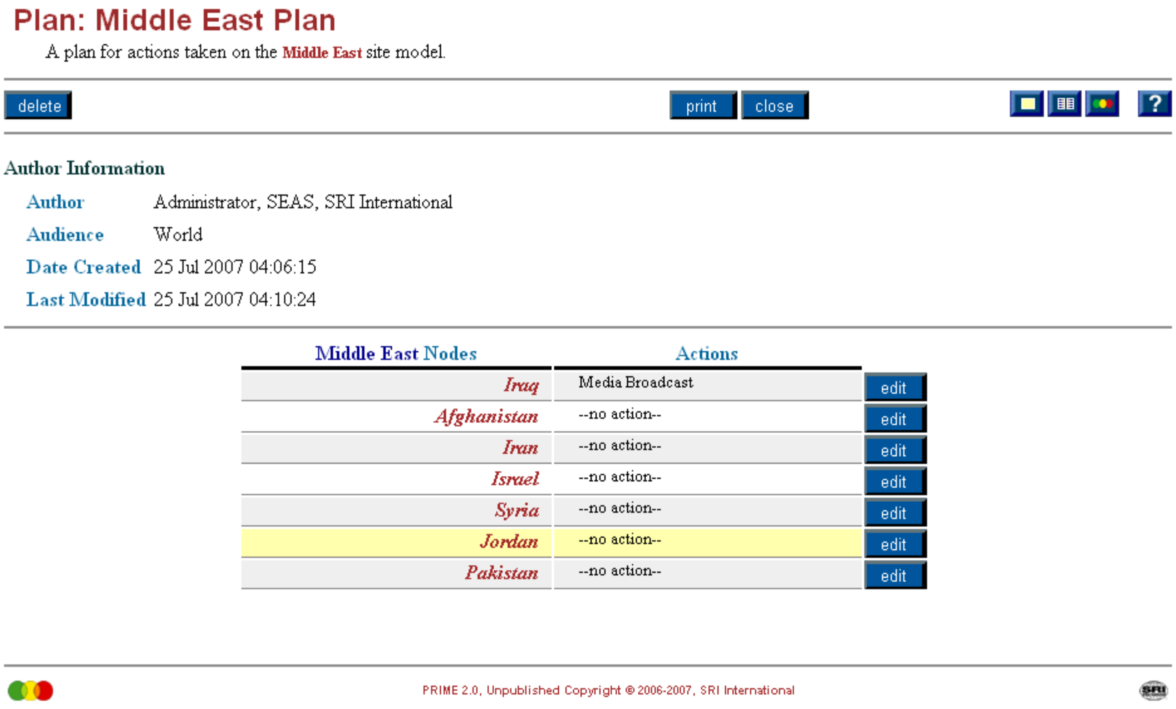
Figure 25: Node Editor

The user continues adding nodes to the site until all the nodes that a plan may consider taking action on have been covered.

4.6.3 Creating a Plan

A plan in PRIME is a set of action types on a set of nodes. The nodes for a plan come from a specified site model, so a plan is always tied to a set site model. Note that the actions are action types rather than actual operational-level actions. Hence a planned action might be “economic embargo (action type) on Iraq (node of node type country)” rather than “Drop JDAMs on building 47”.

The PRIME plan editor is accessed from the main PRIME manager window by clicking on the **Plan** tab. This tab expands to allow the user to either browse all plans to which that user has access (these are displayed in the right frame of the main manager window) or to create a new plan. A user who chooses to create a new plan enters the name of the plan, selects which site model (of those site models the user has access to) the plan applies to, and presses the create button. Otherwise, an existing site model can be opened from the list in the right frame. In either case the plan editor is then displayed (Figure 26).



Plan: Middle East Plan

A plan for actions taken on the **Middle East** site model.

Author Information

Author	Administrator, SEAS, SRI International
Audience	World
Date Created	25 Jul 2007 04:06:15
Last Modified	25 Jul 2007 04:10:24

Middle East Nodes	Actions	
Iraq	Media Broadcast	edit
Afghanistan	--no action--	edit
Iran	--no action--	edit
Israel	--no action--	edit
Syria	--no action--	edit
Jordan	--no action--	edit
Pakistan	--no action--	edit

PRIME 2.0, Unpublished Copyright © 2006-2007, SRI International

Figure 26: Plan Editor

The plan editor has the following functionality in its toolbar:

Delete: Lets a user with write access to the plan delete it

Delete: Prints the plan

Close: Closes the plan editor window

▣**:** Opens memo dialog

▥: Opens access control dialog

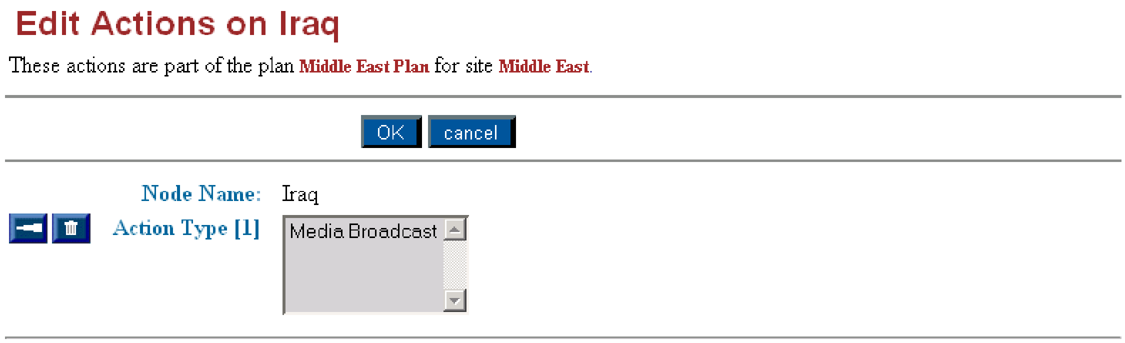
●●●: Brings main PRIME window to foreground

?: Displays help

Below the toolbar is some meta-data about the plan itself (access control, date of creation, last date of modification) and below that is the actual plan editor.

The plan is automatically populated with all the nodes from the chosen site model. The site model is reported below the Plan name and is an active link. When a user clicks on the site model name, the site model editor opens with the selected site model loaded.

For each of the nodes in the site model the user can add/modify/delete action types by clicking on the edit button to open up the plan action editor (Figure 27).



Edit Actions on Iraq

These actions are part of the plan Middle East Plan for site Middle East.

Node Name: Iraq

Action Type [1] Media Broadcast

Figure 27: Plan Action Editor

The user can then click on the ▬ icon to open the action-type taxonomy browser, which contains the action type taxonomy defined by the organization. The user can select zero or more action types to apply to a node. Once completed, the user clicks on the *OK* button, the window closes, and the Plan editor window is updated to display the selected action types for the node (Figure 26). Note that PRIME plans are at a strategic, rather than operational or tactical, level. Hence planned action types for PRIME are generic terms, for example “media broadcast”, rather than something specific like “send videos to all broadcasting companies in Middle East media outlets about success in recruiting police reservists in Anbar province”.

4.7 In-Crisis Use of PRIME

4.7.1 Creating and Editing a Specific PMESII Effects Model

Specific model creation in PRIME starts with a plan and then merges relevant generic models, from the library of known PRIME generic models, to produce the base specific PMESII model for a plan. The user can examine and modify this merged model.

The PRIME specific model editor is accessed from the main PRIME manager window by clicking on the **PMESII Model** tab. This tab expands to allow the user to either browse all PMESII models to which the user has access (these are displayed in the right frame of the main manager window) or to create a new PMESII model. A user who chooses to create a new

PMESII model enters the name of the PMESII model and selects, from a viewable set of plans, a plan to apply and then presses the create button (Figure 28).

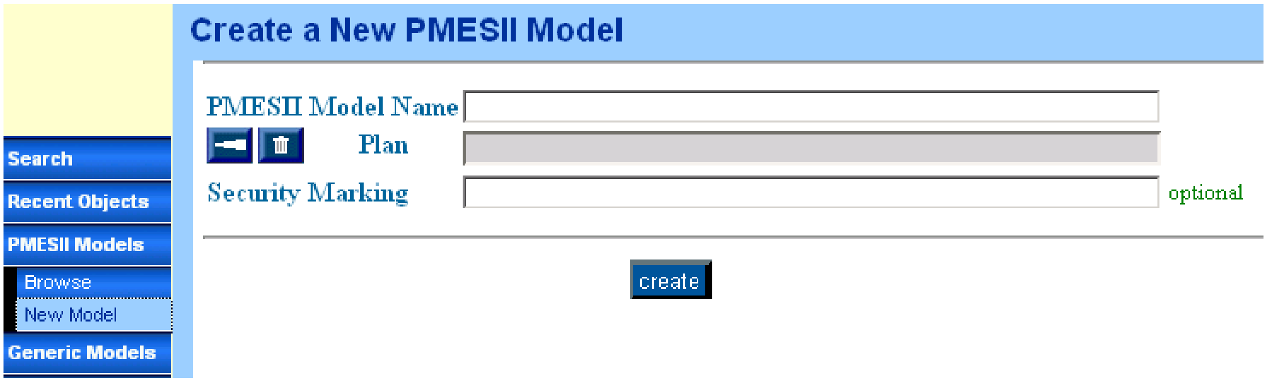
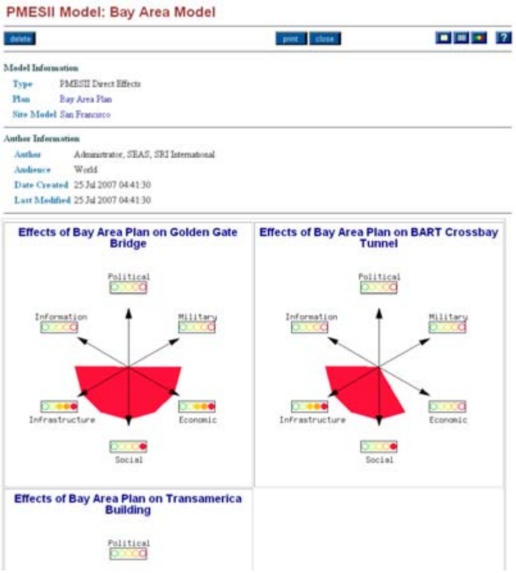


Figure 28: Creating a New Specific PMESII Model

After the create button is pressed, PRIME does reasoning to determine which are the relevant generic models to merge and then merges these models to produce a set of merged PMESII models. The algorithm for deciding on the relevant models to be merged works by taking each plan node in turn and finding generic models that have the same node type (or a subclass of the node's type) defined as the node type of the generic model, and that also have as their action type one that is either the same as a planned action type of the node or a subclass of the planned action. Because all generic models have exactly the same template, SEAS's fusion methods can be used to combine answer values from identical parts of each model to arrive at a combined effects model for the node. SEAS's Bounds data fusion algorithm (see Section 4.6.1) is for data answer fusion.

The result is a collection of PMESII models reflecting the predicted direct effects of a plan, with the collection containing a model for each node in the plan (Figure 29).



PMESII Model: Bay Area Model

Model Information

Type PMESII Direct Effects

Plan Bay Area Plan

Site Model San Francisco

Author Information

Author Administrator, SEAS, SRI International

Audience World

Date Created 25 Jul 2007 04:41:30

Last Modified 25 Jul 2007 04:41:30

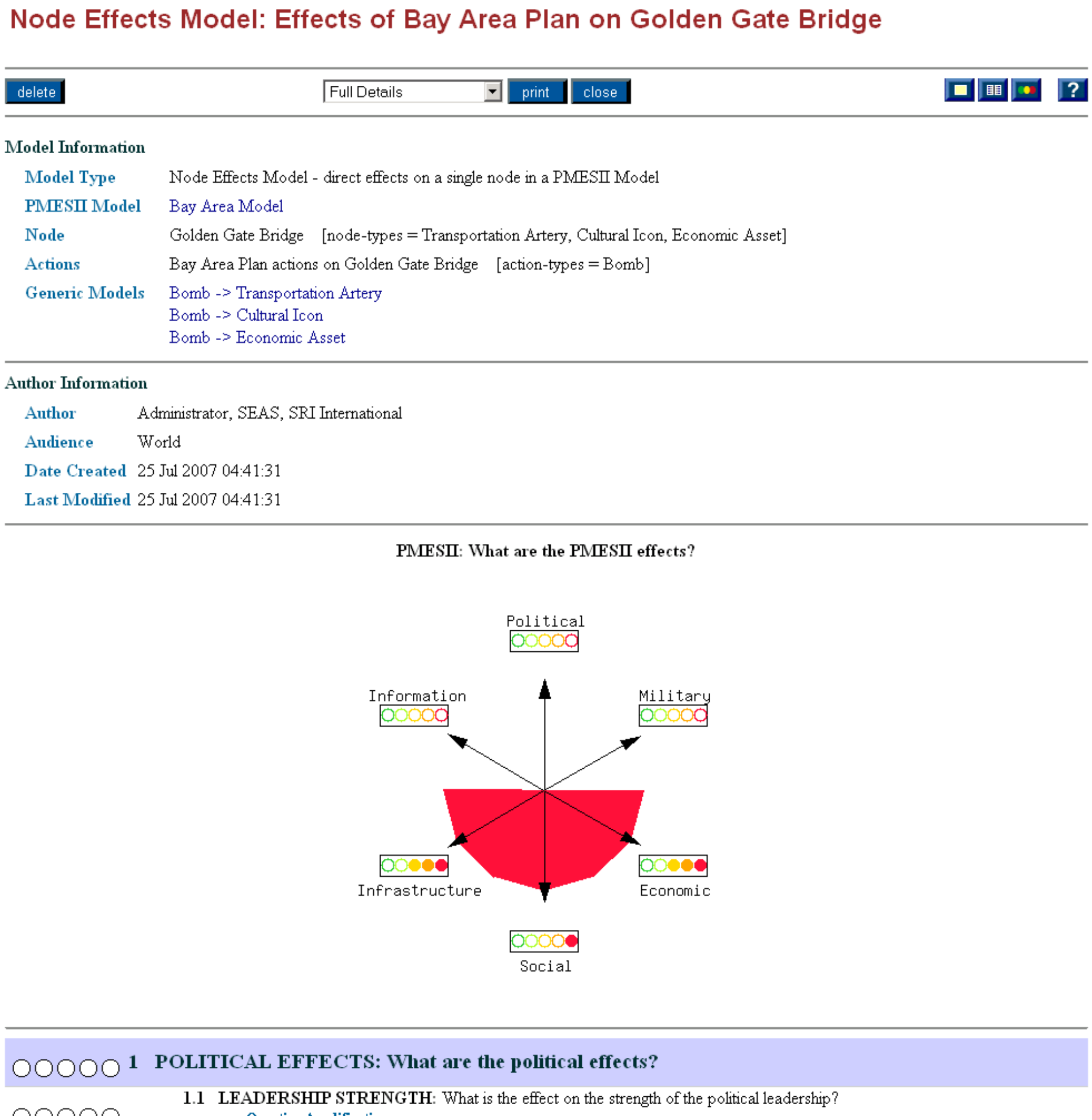
Effects of Bay Area Plan on Golden Gate Bridge

Effects of Bay Area Plan on BART Crossbay Tunnel

Effects of Bay Area Plan on Transamerica Building

Figure 29: Collection of PMESII Models for a Plan

The user can then view each of these PMESII models simply by clicking on one of the models, which opens up the PMESII model editor for a particular node (Figure 30). This PMESII model editor functionality is identical to that of the generic model editor (see Section 4.6.1) and differs only in the reported meta-data. As can be see in Figure 30, the meta-data states which PMESII model collection this model belongs to, the node the model is about, and the actions that the plan calls for on the node. The relevant generic models (as determined by PRIME) that were merged to produce this model are also displayed.



Node Effects Model: Effects of Bay Area Plan on Golden Gate Bridge

Model Information

Model Type	Node Effects Model - direct effects on a single node in a PMESII Model
PMESII Model	Bay Area Model
Node	Golden Gate Bridge [node-types = Transportation Artery, Cultural Icon, Economic Asset]
Actions	Bay Area Plan actions on Golden Gate Bridge [action-types = Bomb]
Generic Models	Bomb -> Transportation Artery Bomb -> Cultural Icon Bomb -> Economic Asset

Author Information

Author	Administrator, SEAS, SRI International
Audience	World
Date Created	25 Jul 2007 04:41:31
Last Modified	25 Jul 2007 04:41:31

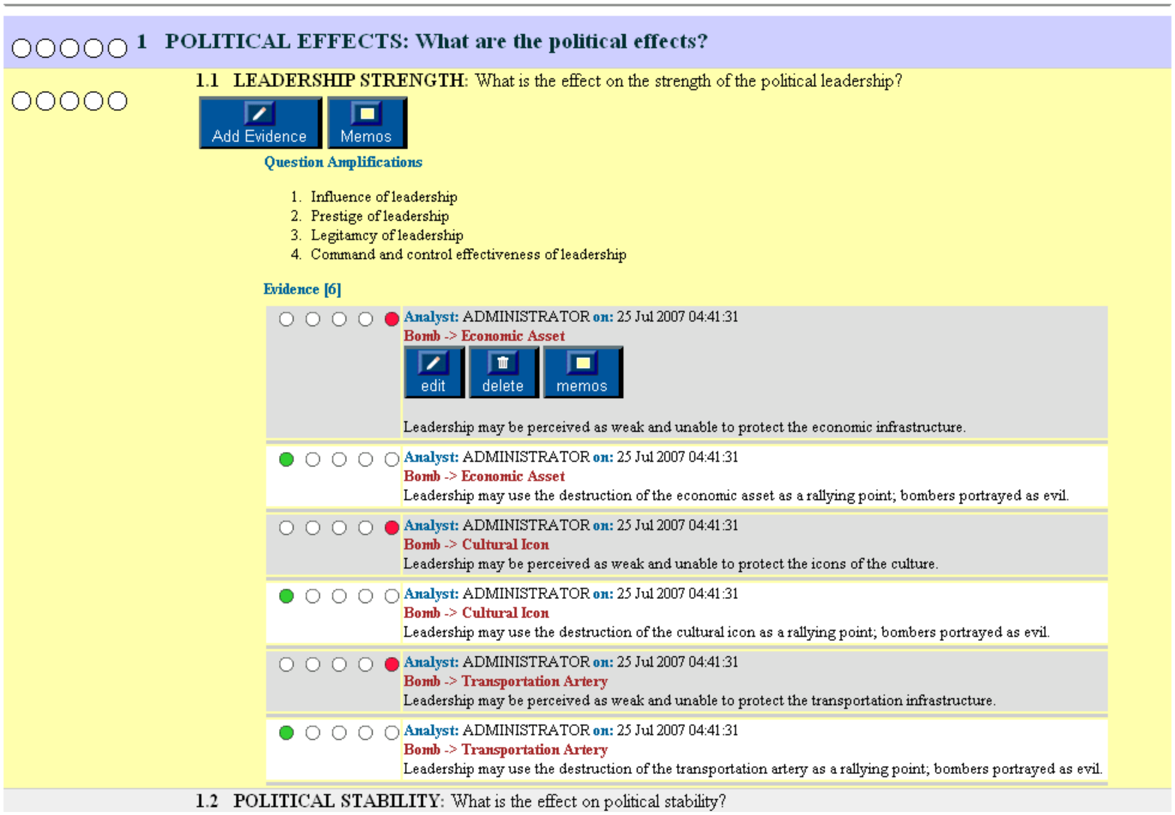
PMESII: What are the PMESII effects?

○○○○○ 1 **POLITICAL EFFECTS: What are the political effects?**

1.1 **LEADERSHIP STRENGTH:** What is the effect on the strength of the political leadership?

Figure 30: PMESII Model Editor for a Plan Node

In the same way as for the generic model editor (see Section 4.6.1), the user who has write permissions for the model can edit this model. PRIME has auto-generated these models, but users examining the models may disagree with what they see. They can view the model, view all the answers that were combined to produce an overall answer at each branch of the model, and delete or modify these answers (Figure 31). Users can also send memos to one another to discuss the results and any edits.



○○○○○ 1 POLITICAL EFFECTS: What are the political effects?

○○○○○ 1.1 LEADERSHIP STRENGTH: What is the effect on the strength of the political leadership?

Question Amplifications

1. Influence of leadership
2. Prestige of leadership
3. Legitamcy of leadership
4. Command and control effectiveness of leadership

Evidence [6]

○ ○ ○ ○ ● Analyst: ADMINISTRATOR on: 25 Jul 2007 04:41:31
Bomb -> Economic Asset
Leadership may be perceived as weak and unable to protect the economic infrastructure.

● ○ ○ ○ ○ Analyst: ADMINISTRATOR on: 25 Jul 2007 04:41:31
Bomb -> Economic Asset
Leadership may use the destruction of the economic asset as a rallying point; bombers portrayed as evil.

○ ○ ○ ○ ● Analyst: ADMINISTRATOR on: 25 Jul 2007 04:41:31
Bomb -> Cultural Icon
Leadership may be perceived as weak and unable to protect the icons of the culture.

● ○ ○ ○ ○ Analyst: ADMINISTRATOR on: 25 Jul 2007 04:41:31
Bomb -> Cultural Icon
Leadership may use the destruction of the cultural icon as a rallying point; bombers portrayed as evil.

○ ○ ○ ○ ● Analyst: ADMINISTRATOR on: 25 Jul 2007 04:41:31
Bomb -> Transportation Artery
Leadership may be perceived as weak and unable to protect the transportation infrastructure.

● ○ ○ ○ ○ Analyst: ADMINISTRATOR on: 25 Jul 2007 04:41:31
Bomb -> Transportation Artery
Leadership may use the destruction of the transportation artery as a rallying point; bombers portrayed as evil.

1.2 POLITICAL STABILITY: What is the effect on political stability?

Figure 31: Editing a Merged PMESII Model

5. Future Research

5.1 Evaluating PRIME

We currently believe that PRIME provides sufficient functionality for it to be of practical use by a team of collaborators (who may be separated in both time and space) to develop rich PMESII models representing the expected direct effects of enacting a plan. What is still to be decided is whether the actual implementation as embodied in PRIME is usable by the analysts who currently carry out this process manually. What is required is an in-depth evaluation of how PRIME supports the PMESII modeling task, which in turn requires analyzing in more detail what constitutes the PMESII modeling task within the Department of Defense (DoD).

This project took as its starting point the ONA process as described in the United States Joint Forces Command *Doctrinal Implications of Operational Net Assessment (ONA)* (http://www.dtic.mil/doctrine/education/jwfc_pam4.pdf). In this document the ONA concept was demonstrated to have strong potential and was recognized as beneficial to military operations. A key part of ONA is the development of a system-of-systems analysis that examines the adversary's PMESII systems. This project lacked input from actual potential end users in the U.S. Air Force (USAF) or other parts of the U.S. Armed Services in order to carry out a thorough knowledge acquisition, use case, and requirements analysis exercise. Our use case and CONOPS was based solely on documents from JFCOM, such as *Doctrinal Implications of Operational Net Assessment (ONA).* Future research would examine the different doctrinal uses of PMESII modeling; examine how, when, and by whom PMESII modeling is conducted; examine the technology support aids that are currently available; and identify where technology support would have the most impact.

In parallel, we envisage conducting experiments with real end users, to identify whether the basic goal of the project – to produce a tool for rapid PMESII model creation – was met. The experiment would not look at usability as its primary metric. PRIME is a prototype and probably has major usability issues that need addressing. Instead, we envisage training users in how to use PRIME, and then the experiment would consider whether PRIME actually supports the process it is meant to support (PMESII modeling) and whether using PRIME speeded up model creation or slowed it down. A secondary criterion would be to measure the quality of the models produced, as judged by experts. Even if PRIME models took longer to create than by using traditional pen-and-paper approaches, they would still have value if the models produced were viewed as being better.

5.2 Higher-order Effect Models

This project intended at the outset to conduct a small experiment in trying to discover indirect (higher-order) PMESII effects, rather than just direct effects that it currently considers. Because of the lack of time and refocus by CPE program management; this part of the overall project was never investigated.

There are many possible effects that direct effect PMESII models may not anticipate. Examples include second- and higher-order effects from an action, e.g., an effect that results from another effect that results from an action, effects that result from combinations of actions and other effects, and effects that result from unique situation features. Discovery of these types of effect is complicated by the fact that the data describing the world will not be perfect, or perfectly modeled. That is, the effect-discovery technique must be tolerant of missing information and some amount of noise in the data, and be able to bridge between general knowledge in the model and specific situations and problems.

To address these issues we would attempt to develop an experimental Higher-Order Effects Discovery Tool. The tool would be based on SRI's Link Analysis Workbench (LAW)[5] tool,

[5] LAW – http://www.ai.sri.com/~law

developed under the ARDA EAGLE programs. At LAW's core is a pattern-matching engine that (1) provides an intuitive graphical representation of patterns and matches, (2) allows "neighborhood" matches to the pattern via a graph edit distance similarity metric, (3) has a simple graphical pattern editor that allows analysts to create their own patterns, and (4) supports higher-level constructs in the patterns – hierarchy, disjunction, recursion, and so on – that allow representation of complex scenarios. A library of LAW patterns would be used to represent the interactions between various elements in the data that predict effects – interactions between actions, urban infrastructure, groups, and so on – and the matcher would find matching or partially matching instances in the data and feed them (along with strength-of-match measures) to PRIME for further analysis. Transforming LAW into a full-fledged higher-order effects detection tool would involve addressing two key technical challenges:

- *Building an Effects Pattern Knowledge Base.* To discover higher-order effects, LAW needs a library of patterns that define the scenarios under which these effects occur. We would develop simple example patterns initially, sufficient for subsequent experimentation. The existing LAW pattern editing software is sufficient for this purpose. Even further in the future we could envisage how subject matter experts could create a rich library of patterns.
- *Communicating Predicted Effects.* LAW results need to be entered as evidence for PRIME PMESII models.

6. Conclusions

Overall we believe that the PRIME project was successful and that the developed PRIME software is a practical tool for rapidly developing PMESII models. While PRIME has not been evaluated against a realistic evaluation scenario, the fact remains that PRIME can support collaborative development of PMESII models, as we have tested among our internal team in order to produce demonstration example PMESII models, which are part of the PRIME distribution. To our knowledge, no other existing tool for PMESII modeling supports collaboration, nor does it allow for such rapid PMESII model development. Indeed, paper-based methods are reported to be still in wide use.

During the project, many design decisions were taken to streamline the process of reaching the end goal – the production of PMESII direct effect models. We went further in the tailoring of the PRIME plug-in than we had originally envisaged, suppressing more of the underlying SEAS GUI and functionality, with the goal of hiding distracting clutter and information from the user. In particular, the redesigned main window for PRIME is radically different from the main window of SEAS and provides a PMESII modeling-centric view of the world.

Some design decisions, such as severely limiting the editing of the underlying PMESII model template, were made late in the project, as the full ramifications of allowing a lot of degrees of freedom in template design were found to lead to potentially serious downstream issues when developing PMESII models. In the end, the decision was made to produce a tool that would lead to consistent and understandable results and would not lead to misunderstanding, such as when the underlying model templates were changed after models had been built. In an organizational setting, business process rules should handle many of these issues. However, we felt it prudent to limit the software's features now and, if requested by using organizations, to add back certain functionality.

PRIME has been delivered to AFRL and is under evaluation by an external contractor on behalf of AFRL for the CPE program. Feedback has already identified some terminological and technical issues. Probably the most important is that PRIME does not run correctly under Internet Explorer, which is due to problems in PRIME's JavaScript GUI code. SRI intends to invest in and develop PRIME further, driven by user feedback, in order to make PRIME a better PMESII modeling tool. To this end SRI has stood up a live PRIME server and welcomes requests by interested parties for accounts on this server (send a request to seas@ai.sri.com).

List of Acronyms

AFRL – Air Force Research Laboratory
ARDA – Advanced Research and Development Activity
ALP – Active Lisp Pages
API – Application Programming Interface
BOGSAT – Bunch Of Guys Sitting Around a Table
CONOPS – Concept of Operations
COTS – Commercial Off The Shelf
CPE – Commander’s Predictive Environment
DBMS – DataBase Management System
DIME – Diplomatic, Information, Military, Economic
DNS – Domain Name Server
DoD – Department of Defense
EAGLE – Evidence Assessment, Grouping, Linking and Evaluation
EBO – Effects-Based Operations
GOTS – Government Off The Shelf
HTML – Hyper-Text Markup Language
HTTP – Hyper-Text Transfer Protocol
IC – Intelligence Community
IE – Internet Explorer
IP – Internet Protocol
ISX – A technology company, not an acronym.
JDAM – Joint Direct Attack Munition
JFCOM – Joint Forces Command
LAW – Link Analysis Workbench
KB – Knowledge Base
KBMS – Knowledge Based Management System
MNE02 – Multi-National Experiment ‘02
OKBC – Open Knowledge Base Connectivity
ONA – Operational Net Assessment
OS – Operating System
OWL – Ontology Web Language
PMESII – Political, Military, Economic, Social, Information, Infrastructure
PRIME – PMESII Rapid Interactive Modeling Environment
SEAS – Structured Evidential Argumentation System
SQL – Standard Query Language
SRI – A technology company, not an acronym.
URL – Uniform Resource Locator
USAF – United States Air Force